Knowledge-Based Systems 238 (2022) 107886

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

A robust variational autoencoder using beta divergence

Haleh Akrami [a,*], Anand A. Joshi [a], Jian Li [b,c], Sergül Aydöre [d], Richard M. Leahy [a]

[a] Signal and Image Processing Institute, University of Southern California, Los Angeles, CA, USA
[b] Athinoula A. Martinos Center for Biomedical Imaging Massachusetts General Hospital, Harvard Medical School, Charlestown, MA, USA
[c] Center for Neurotechnology and Neurorecovery, Department of Neurology, Massachusetts General Hospital and Harvard Medical School, Boston, MA, USA
[d] Amazon Web Services, New York, NY, USA

ARTICLE INFO

Article history:
Received 30 July 2021
Received in revised form 21 October 2021
Accepted 2 December 2021
Available online 10 December 2021

Keywords:
RVAE
Robust anomaly detection
Outlier
VAE
β divergence

ABSTRACT

The presence of outliers can severely degrade learned representations and performance of deep learning methods and hence disproportionately affect the training process, leading to incorrect conclusions about the data. For example, anomaly detection using deep generative models is typically only possible when similar anomalies (or outliers) are not present in the training data. Here we focus on variational autoencoders (VAEs). While the VAE is a popular framework for anomaly detection tasks, we observe that the VAE is unable to detect outliers when the training data contains anomalies that have the same distribution as those in test data. In this paper we focus on robustness to outliers in training data in VAE settings using concepts from robust statistics. We propose a variational lower bound that leads to a robust VAE model that has the same computational complexity as the standard VAE and contains a single automatically-adjusted tuning parameter to control the degree of robustness. We present mathematical formulations for robust variational autoencoders (RVAEs) for Bernoulli, Gaussian and categorical variables. The RVAE model is based on beta-divergence rather than the standard Kullback–Leibler (KL) divergence. We demonstrate the performance of our proposed β-divergence-based autoencoder for a variety of image and categorical datasets showing improved robustness to outliers both qualitatively and quantitatively. We also illustrate the use of our robust VAE for detection of lesions in brain images, formulated as an anomaly detection task. Finally, we suggest a method to tune the hyperparameter of RVAE which makes our model completely unsupervised.

1. Introduction

learning models that are based on the maximization of the log-likelihood (e.g. autoencoders) often assume perfect training data [1]. Outliers in training data can have a disproportionate impact on learning because they have large negative log-likelihood values for a correctly trained network [2,3]. In practice, particularly in large datasets, training data will inevitably include mislabeled data, anomalies or outliers, sometimes taking up as much as 10% of the data [4].

In the case of autoencoders, the inclusion of outliers in the training data can result in the encoding of these outliers. As a result, the trained network may reconstruct these outliers in the testing samples. Conversely, if an encoder is robust to outliers then the outliers will not be reconstructed correctly. Therefore, a robust autoencoder can be used to detect anomalies that are presented at inference time by comparing an original image to its reconstructed version.

Here, we focus on variational autoencoders (VAEs) [5]. A VAE is a probabilistic graphical model that is comprised of an encoder and a decoder. The encoder transforms high-dimensional input data with an intractable probability distribution into a low-dimensional 'code' with an approximate tractable posterior (variational distribution). The decoder takes the code as an input and returns the parameters of the conditional distribution of the data. VAEs use the concept of variational inference [6] and reparameterize the variational evidence lower bound (ELBO) so that it can be optimized using standard stochastic gradient descent methods.

A VAE can learn latent features that best describe the distribution of the data and allows the generation of new samples using the decoder. VAEs have been successfully used for feature extraction from images, audio and text [7–9]. As noted above, when VAEs are trained using normal datasets, they can be used to detect anomalies, where the characteristics of the anomalies differ from those of the training data [10,11]. It has been shown that the variational form is preferable to standard autoencoders

* Corresponding author.
E-mail addresses: akrami@usc.edu (H. Akrami), ajoshi@usc.edu (A.A. Joshi), jli112@mgh.harvard.edu (J. Li), sergulaydore@gmail.com (S. Aydöre), leahy@sipi.usc.edu (R.M. Leahy).

https://doi.org/10.1016/j.knosys.2021.107886
0950-7051/

in real-world applications such as lesion detection [12–14]. For example, Chen et al. [15] reformulated pixel-wise lesion detection as an image restoration problem using a VAE as a probabilistic model with a network-based prior as the normative distribution resulting in a reduced false positive rate. A recent paper [16] suggested cautious use of VAE likelihood for anomaly detection as they show VAE might assign a high likelihood for the out-of-distribution samples in some settings. Importantly, they base their conclusions on the ability to detect anomalies on differences in likelihood between inliers and outliers. Here we use the pixel-wise reconstruction error measure for anomaly detection.

The VAE effectively assigns probabilities (or likelihoods) to the data. Since deep generative models are very flexible, they are able to over-fit to outliers that are present in the training data. This in turn will result in high probabilities for outliers in the inference time [17]. If the goal is to detect outliers as anomalies characterized as not being accurately encoded by the VAE, this propensity for over-fitting will negatively impact performance. For this reason, we develop a Robust VAE (RVAE) framework that is robust to the presence of outliers in the training data. In other words, these outliers are encoded with low probability in the trained network. We note that in contrast to the case treated in here, the presence of outliers during training can be used to improve performance in unsupervised models when the uncorrupted versions of the samples is also known, which is the core idea of the Denoising autoencoder (AE) [18].

It is worth mentioning that definition of robustness is context dependent. Here we are specifically interested in *insensitivity to deviations from underlying assumptions in the training data*. As a concrete example, in the primarily application presented later we are interested in detecting lesions in magnetic resonance images of the brain. Our underlying assumption is that there are no lesions present in the training images. Outliers are defined as images that do contain lesions. Our goal is to train our RVAE so that even if some of the training data do contain lesions, these outliers are poorly encoded so that we still are able to detect lesions using the trained network by comparing the original and decoded images. Here we show that the presence of outliers in training data can result in degraded performance of VAEs for anomaly detection. We then describe a robust VAE that overcomes this problem.

1.1. Related work

In the past few years, denoising autoencoders [18], maximum correntropy autoencoders [19] and robust autoencoder [20] have been proposed to overcome the problem of noise corruption, anomalies, and outliers in the data. The denoising autoencoder [18] is trained to reconstruct 'noise-free' inputs by corrupting the input data during training and is robust to the type of corruption it learns. However, these denoising autoencoders require access to clean training data and the modeling of noise can be difficult in real-world problems. An alternative approach is to replace the cost function with noise-resistant correntropy [19]. Although this approach discourages the reconstruction of the outliers in the output, it may not prevent from encoding of outliers in the hidden layer. Recently, Zhou and Paffenroth [20] described a robust deep autoencoder that was inspired by robust principal component analysis. This encoder performs a decomposition of input data $\mathbf{X}$ into two components, $\mathbf{X} = \mathbf{LD} + \mathbf{S}$, where $\mathbf{LD}$ is the low-rank component which we want to reconstruct and $\mathbf{S}$ represents a sparse component that contains outliers or noise. Despite many successful applications of these models, they do not extend well to generative models and categorical datasets.

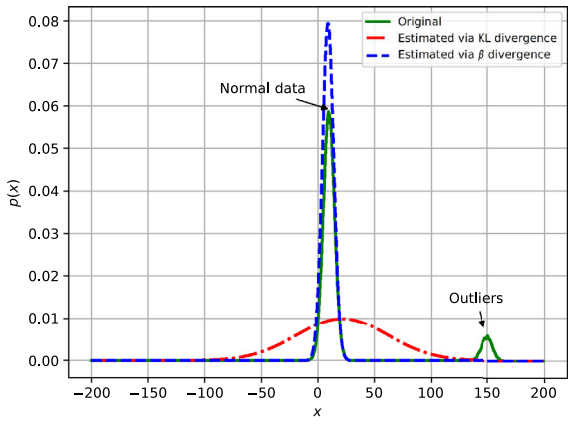

Fig. 1. Illustration of the robustness of β-divergence to outliers in comparison to KL-divergence: optimizing KL-divergence for parameter estimation of a single Gaussian distribution does not distinguish between inliers and outliers, whereas optimizing β-divergence results in an estimate that is robust to outliers, by de-emphasizing out of distribution data.

1.2. Motivation via a toy example

To make the VAE models robust to outliers in the training data, existing approaches focus on the modification of network architectures, adding constraints or modeling of outlier distributions [17,21,22]. In contrast, we adopt an approach based on the β-divergence from robust statistics [23]. To motivate our approach we ran the simulation illustrated in Fig. 1. Here, the samples are generated from a distribution p which is a mixture of two Gaussian distributions where the tall mode represents inlier samples and the short mode indicates the presence of outliers. Our goal is to learn a single-mode Gaussian distribution p_θ by minimizing either the Kullback–Leibler (KL) or β divergences to optimize parameters θ. β-divergence is defined as D_β [23]:

$$D_\beta(p(\mathbf{X}) \parallel p_\theta(X)) = \frac{1}{\beta}\int_X p(\mathbf{X})^{\beta+1} d\mathbf{X} - \frac{\beta+1}{\beta}\int_{\mathbf{X}} p(\mathbf{X}) p_\theta(\mathbf{X})^\beta d\mathbf{X} + \int_{\mathbf{X}} p_\theta(\mathbf{X})^{\beta+1} d\mathbf{X}.$$

Fig. 1 shows the estimated distributions found by using the two different divergences measures. While the β-divergence estimate is robust to the outliers, the estimated Gaussian distribution from the KL divergence attempts to also account for their presence and misplaces the mean and variance of the estimated distribution. We observed similar results over a range of variances for inlier and outlier distributions: VAE estimates were consistently influenced by both distributions while RVAE learned only the (dominant) inlier distribution

In contrast to our toy example, in practice p(X) is unknown and is replaced with an empirical distribution. To learn this distribution using the standard KL-based approach we minimize: $\arg\min_\theta D_{KL}(\hat{p}(\mathbf{X}) \parallel p_\theta(\mathbf{X})) = Const - \frac{1}{N}\sum_{i=1}^{N} \log(p_\theta(\mathbf{x^{(i)}}))$ where N is the number of samples and p_θ is the estimate of the empirical distribution. This is a maximum likelihood estimation (MLE) problem that is sensitive to outliers because it treats all data points equally. Replacing KL with a density power divergence can overcome this problem, with β-divergence a popular choice. Minimizing β-divergence is equivalent to minimizing β-cross entropy defined as [24]:

$$H_\beta(p(\mathbf{X}), p_\theta(\mathbf{X})) = - \frac{\beta+1}{\beta}\int \hat{p}(\mathbf{X})(p_\theta(\mathbf{X})^\beta - 1) d\mathbf{X} + \int p_\theta(\mathbf{X})^{\beta+1} d\mathbf{X}. \quad (1)$$

This simplifies for empirical estimation to:

$$L_\beta(\theta) = Const - \frac{\beta+1}{\beta}\sum_{i=1}^{N} p_\theta(\mathbf{x^{(i)}})^\beta + E_{p_\theta(\mathbf{x})}[p_\theta(\mathbf{X})^\beta]$$

Setting the derivative with respect to θ to zero results in:

$$0 = -\sum_{i=1}^{N} p_\theta(\mathbf{x^{(i)}})^\beta \frac{\partial}{\partial\theta}\log(p_\theta(\mathbf{x^{(i)}})) + E_{p_\theta(\mathbf{x})}[p_\theta(\mathbf{X})^\beta \frac{\partial}{\partial\theta}\log(p_\theta(\mathbf{X}))]$$

The first term is the likelihood weighted according to the β-power of the probability density for each data point. This equation weights inliers (high probability samples) more than the outliers (low probability samples) since the probability densities are higher for the former. Consequently, $L_\beta(\theta)$ suppresses the likelihood of outliers and can be interpreted as an M-estimate [23]. Inspired by this formulation we derive a new reconstruction error term for VAE using β divergence. We used the influence function (IF) [25] for robustness analysis of our new VAE as described in Section 2.3.4.

We note that the standard VAE has some inherent robustness and is related to Robust PCA [26]. However, we show in experiments below that VAE's performance degrades with the increasing presence of outliers in training. Our formulation is a generalization of VAE which converges to it as $\beta \to 0$ (Appendix) and adds extra robustness. Note that the RVAE formulation works with any VAE setup (even when the constant posterior variance goes to zero and the VAE converges to AE.

1.3. Our contributions

We propose a novel robust VAE (RVAE) using robust variational inference [25] that uses a $\beta-$ELBO-based cost function. The β-ELBO cost replaces the KL-divergence (log-likelihood) term with β-divergence. Our contributions are as follows:

- We apply concepts from robust statistics, specifically, robust variational inference to variational autoencoder (VAE) for deriving a robust variational autoencoder (RVAE) model. We also present formulations of RVAE for Gaussian and Bernoulli models as well as categorical and mixed type data.
- We show that on datasets from computer vision, network traffic, and real-world brain imaging that our approach is more robust than a standard VAE to outliers.
- We also show how the robustness of RVAE can be exploited to perform anomaly detection even in cases where the training data also includes similar anomalies. We performed anomaly detection both for images and tabular data sets with categorical and continuous features.
- Finally we suggest an approach for hyperparameter tuning which makes our model completely unsupervised.

2. Mathematical formulation

Let $\mathbf{x}^{(i)} \in \mathbb{R}^D$ be an observed sample of input $\mathbf{X}$ where $i \in \{1,\ldots,N\}$, D is the number of features and N is the number of samples; and $\mathbf{z}^{(j)}$ be an observed sample for latent variable $\mathbf{Z}$ where $j \in \{1,\ldots,S\}$. Given samples $\mathbf{x}^{(i)}$ of the random feature vector $\mathbf{X}$ representing input data, probabilistic graphical models estimate the posterior distribution $p_\theta(\mathbf{Z}|\mathbf{X})$ as well as the model evidence $p_\theta(\mathbf{X})$, where $\mathbf{Z}$ represents the latent variables and θ the generative model parameters [6]. The goal of variational inference is to approximate the posterior distribution of $\mathbf{Z}$ given $\mathbf{X}$ by a tractable parametric distribution. In variational methods, the functions used as prior and posterior distributions are restricted to those that lead to tractable solutions. For any choice of a tractable $q(\mathbf{Z})$, the distribution of the latent variable, the following decomposition holds:

$$\log p_\theta(\mathbf{X}) = L(q(\mathbf{Z}),\theta) + D_{KL}(q(\mathbf{Z}) \parallel p_\theta(\mathbf{Z}|\mathbf{X})), \tag{2}$$

where:

$$L(q(\mathbf{Z},\theta)) = \mathbb{E}_{q(\mathbf{Z})}[\log(p_\theta(\mathbf{X}|\mathbf{Z}))] - D_{KL}(q(\mathbf{Z}) \parallel p_\theta(\mathbf{Z})).$$

and D_{KL} represents the Kullback–Leibler (KL) divergence. Instead of maximizing the log-likelihood $p_\theta(\mathbf{X})$, with respect to the model parameters θ, the variational inference approach maximizes its variational evidence lower bound ELBO [6].

2.1. Robust variational inference

Here we review the robust variational inference framework [25] and explain its usage for developing variational autoencoders that are robust to outliers. The ELBO function includes a log-likelihood term which is sensitive to outliers in the data because the negative log-likelihood of low probability samples can be arbitrarily high. It can be shown that maximizing log-likelihood given samples $\mathbf{x}^{(i)}$ is equivalent to minimizing KL divergence $D_{KL}(\hat{p}(\mathbf{X}) \parallel p_\theta(\mathbf{X}|\mathbf{Z}))$ between the empirical distribution $\hat{p}$ of the samples and the parametric distribution p_θ [25,27]. Therefore, the ELBO function can be expressed as:

$$\begin{aligned} L(q,\theta) = &- N\mathbb{E}_q\left[D_{KL}(\hat{p}(\mathbf{X}) \parallel p_\theta(\mathbf{X}|\mathbf{Z}))\right] \\ &- D_{KL}(q(\mathbf{Z}) \parallel p_\theta(\mathbf{Z})) + \text{const.}, \end{aligned} \tag{3}$$

where N is the number of samples of $\mathbf{X}$ used for computing the empirical distribution $\hat{p}(\mathbf{X}) = \frac{1}{N}\sum_{i=1}^{N}\delta(\mathbf{X},\mathbf{x}^{(i)})$ and δ is the Dirac delta function. For the robust case we replace KL with β divergence, D_β [23]:

$$D_\beta(\hat{p}(\mathbf{X}) \parallel p_\theta(\mathbf{X}|\mathbf{Z})) = \frac{1}{\beta}\int_X \hat{p}(\mathbf{X})^{\beta+1}d\mathbf{X} - \frac{\beta+1}{\beta}\int_{\mathbf{X}} \hat{p}(\mathbf{X})p_\theta(\mathbf{X}|\mathbf{Z})^\beta d\mathbf{X} + \int_{\mathbf{X}} p_\theta(\mathbf{X}|\mathbf{Z})^{\beta+1}d\mathbf{X}.$$

In the limit as $\beta \to 0$, D_β converges to D_{KL}. Using β-divergence changes the variational inference optimization problem to maximizing β-ELBO:

$$\begin{aligned} L_\beta(q,\theta) = &- N\mathbb{E}_q[D_\beta(\hat{p}(\mathbf{X}) \parallel p_\theta(\mathbf{X}|\mathbf{Z}))] \\ &- D_{KL}(q(\mathbf{Z}) \parallel p_\theta(\mathbf{Z})) \end{aligned} \tag{4}$$

Note that for robustness to outliers in the input data, the divergence in the likelihood is replaced, but divergence in the latent space is unchanged [25]. The idea behind β-divergence is based on applying a power transform to variables with heavy tailed distributions [28]. It can be proven that minimizing $D_\beta(\hat{p}(\mathbf{X}) \parallel p_\theta(\mathbf{X}|\mathbf{Z}))$ is equivalent to minimizing β-cross entropy [25] and is given by [24]:

$$H_\beta(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z})) = -\frac{\beta+1}{\beta}\int \hat{p}(\mathbf{X})(p_\theta(\mathbf{X}|\mathbf{Z})^\beta - 1)d\mathbf{X} + \int p_\theta(\mathbf{X}|\mathbf{Z})^{\beta+1}d\mathbf{X}. \tag{5}$$

Replacing D_β in Eq. (4) with H_β results in

$$\begin{aligned} L_\beta(q,\theta) = &- N\mathbb{E}_q[H_\beta(\hat{p}(\mathbf{X}) \parallel p_\theta(\mathbf{X}|\mathbf{Z}))] \\ &- D_{KL}(q(\mathbf{Z}) \parallel p_\theta(\mathbf{Z})). \end{aligned} \tag{6}$$

2.2. Variational autoencoder

A variational autoencoder (VAE) is a directed probabilistic graphical model whose posteriors are approximated by a neural network. It has two components: the encoder network that computes $q_\phi(\mathbf{Z}|\mathbf{X})$, which is a tractable approximation of the

intractable posterior $p_\theta(\mathbf{Z}|\mathbf{X})$, and the decoder network that computes $p_\theta(\mathbf{X}|\mathbf{Z})$, which together form an autoencoder-like architecture [29]. The regularizing assumption on the latent variables is that the marginal $p_\theta(\mathbf{Z})$ is a standard Gaussian $\mathcal{N}(0, 1)$. For this model the marginal likelihood of individual data points can be rewritten as follows:

$$\begin{aligned}\log p_\theta(\mathbf{x}^{(i)}) =& D_{KL}(q_\phi(\mathbf{Z}|\mathbf{x}^{(i)}), p_\theta(\mathbf{Z}|\mathbf{x}^{(i)})) \\ &+ L(\theta, \phi; \mathbf{x}^{(i)}),\end{aligned} \tag{7}$$

where

$$\begin{aligned}L(\theta, \phi; \mathbf{x}^{(i)}) =& \mathbb{E}_{q_\phi(\mathbf{Z}|\mathbf{x}^{(i)})}[\log(p_\theta(\mathbf{x}^{(i)}|\mathbf{Z}))] \\ &- D_{KL}(q_\phi(\mathbf{Z}|\mathbf{x}^{(i)}) \parallel p_\theta(\mathbf{Z})).\end{aligned} \tag{8}$$

The first term (log-likelihood) can be interpreted as the *reconstruction loss* and the second term (KL divergence) as the *regularizer*. Using empirical estimates of expectation we form the Stochastic Gradient Variational Bayes (SGVB) cost [5]:

$$\begin{aligned}L(\theta, \phi; \mathbf{x}^{(i)}) \approx& \frac{1}{S}\sum_{j=1}^{S}\log(p_\theta(\mathbf{x}^{(i)}|\mathbf{z}^{(j)})) \\ &- D_{KL}(q_\phi(\mathbf{Z}|\mathbf{x}^{(i)}) \parallel p_\theta(\mathbf{Z})),\end{aligned} \tag{9}$$

where S is the number of samples drawn from $q_\phi(\mathbf{Z}|\mathbf{X})$. We can assume either a multivariate i.i.d. Gaussian or Bernoulli distribution for $p_\theta(\mathbf{X}|\mathbf{Z})$. That is, given the latent variables, the uncertainty remaining in $\mathbf{X}$ is i.i.d. with these distributions. For the Bernoulli case, the log likelihood for sample $\mathbf{x}^{(i)}$ simplifies to:

$$\begin{aligned}\mathbb{E}_{q_\phi}(\log p_\theta(\mathbf{x}^{(i)}|\mathbf{Z})) &\approx \frac{1}{S}\sum_{j=1}^{S}\log p_\theta(\mathbf{x}^{(i)}|\mathbf{z}^{(j)}) \\ &= \frac{1}{S}\sum_{j=1}^{S}\sum_{d=1}^{D}\left(\mathbf{x}^{(i)}\log p_d^{(j)} + (1-\mathbf{x}_d^{(i)})\log(1-p_d^{(j)})\right),\end{aligned}$$

where $p_\theta(\mathbf{x}_d^{(i)}|\mathbf{z}^{(j)}) = \text{Bernoulli}(p_d^{(j)})$ and D is the feature dimension. In practice we can choose $S = 1$ as long as the minibatch size is large enough. For the Gaussian case, this term simplifies to the squared-error when the variance is fixed.

2.3. Robust variational autoencoder

We now derive the robust VAE (RVAE) using concepts discussed above. In order to derive the cost function for the RVAE, as in Eq. (6), we propose to use β-cross entropy $H_\beta^{(i)}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z}))$ between the empirical distribution of the data $\hat{p}(\mathbf{X})$ and the probability of the samples for the generative process $p_\theta(\mathbf{X}|\mathbf{Z})$ for each sample $\mathbf{x}^{(i)}$ in place of the likelihood term in Eq. (9). Similar to VAE, the regularizing assumption on the latent variables is that the marginal $p_\theta(\mathbf{Z})$ is normal Gaussian $\mathcal{N}(0, 1)$. The β-ELBO for the RVAE is:

$$\begin{aligned}L_\beta(\theta, \phi; \mathbf{x}^{(i)}) =& -\mathbb{E}_{q_\phi(\mathbf{Z}|\mathbf{x}^{(i)})}[(H_\beta^{(i)}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z})))] \\ &- D_{KL}(q_\phi(\mathbf{Z}|\mathbf{x}^{(i)}) \parallel p_\theta(\mathbf{Z})).\end{aligned}$$

2.3.1. Bernoulli case

The Bernoulli case is used when the data are binary. For each sample, we need to calculate $H_\beta^{(i)}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z}))$ where $\mathbf{x}^{(i)} \in \{0, 1\}$. Using empirical estimates of expectation we form the SGVB cost and chose S=1. In Eq. (5), we substitute $\hat{p}(\mathbf{X}) = \delta(\mathbf{X} - \mathbf{x}^{(i)})$ and $p_\theta(\mathbf{X}|\mathbf{z}^{(j)})$ is a Bernoulli distribution therefore $p_\theta(\mathbf{X}|\mathbf{z}^{(j)})^\beta = \left(\mathbf{X}p^{(j)} + (1-\mathbf{X})(1-p^{(j)})\right)^\beta = \mathbf{X}p^{(j)\beta} + (1-\mathbf{X})\left(1-p^{(j)}\right)^\beta$, and

$$\begin{aligned}&H_\beta^{(i)}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{z}^{(j)})) = \\ &-\frac{\beta+1}{\beta}\sum_{\mathbf{x}}\delta(\mathbf{X}-\mathbf{x}^{(i)})(\mathbf{X}p^{(j)} + (1-\mathbf{X})(1-p^{(j)})^\beta - 1) \\ &+ p^{(j)\beta+1} + (1-p^{(j)})^{\beta+1}.\end{aligned}$$

We calculate the sum over $\mathbf{x}^{(i)} \in \{0, 1\}$. Therefore, for the multivariate case, the β-ELBO of RVAE becomes:

$$\begin{aligned}&L_\beta(\theta, \phi; \mathbf{x}^{(i)}) = \\ &\frac{\beta+1}{\beta}\left(\prod_{d=1}^{D}\left(\mathbf{x}_d^{(i)}p_d^{(j)\beta} + (1-\mathbf{x}_d^{(i)})(1-p_d^{(j)})^\beta\right) - 1\right) \\ &-\prod_{d=1}^{D}\left(p_d^{(j)\beta+1} + (1-p_d^{(j)\beta+1})\right) - D_{KL}(q_\phi(\mathbf{Z}|\mathbf{x}^{(i)}) \parallel p_\theta(\mathbf{Z})).\end{aligned} \tag{10}$$

2.3.2. Gaussian case

When the data is continuous and unbounded we can assume the posterior $p(\mathbf{X}|\mathbf{z}^{(j)})$ is Gaussian $\mathcal{N}(\hat{\mathbf{x}}^{(j)}, \sigma)$ where $\hat{\mathbf{x}}^{(j)}$ is the output of the decoder generated from $\mathbf{z}^{(j)}$. Here, we choose $\sigma = 0.5$ for our experiments since values were normalized between 0 and 1. We empirically found that we can make VAE robust for any value of σ. The β-cross entropy for the ith sample input $\mathbf{x}^{(i)}$ is given by:

$$\begin{aligned}&H_\beta^{(i)}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{z}^{(j)})) = \\ &-\frac{\beta+1}{\beta}\int\delta(\mathbf{X}-\mathbf{x}^{(i)})(N(\hat{\mathbf{x}}^{(j)}, \sigma)^\beta - 1)d\mathbf{X} \\ &+\int N(\hat{\mathbf{x}}^{(j)}, \sigma)^{\beta+1}d\mathbf{X}.\end{aligned} \tag{11}$$

The second term does not depend on $\hat{\mathbf{x}}$ so the first term is minimized when $\exp(-\beta\sum_{d=1}^{D}\|\hat{\mathbf{x}}_d^{(j)} - \mathbf{x}_d^{(i)}\|^2)$ is maximized. Therefore, the ELBO-cost for the Gaussian case for jth sample is then given by

$$\begin{aligned}&L_\beta(\theta, \phi; \mathbf{x}^{(i)}) = \\ &\frac{\beta+1}{\beta}\left(\frac{1}{(2\pi\sigma^2)^{\beta D/2}}\exp\left(-\frac{\beta}{2\sigma^2}\sum_{d=1}^{D}\|\hat{\mathbf{x}}_d^{(j)} - \mathbf{x}_d^{(i)}\|^2\right) - 1\right) \\ &- D_{KL}(q_\phi(\mathbf{Z}|\mathbf{x}^{(i)}) \parallel p_\theta(\mathbf{Z})).\end{aligned} \tag{12}$$

The cost converges to MSE in the limiting case $\beta \to 0$ (Appendix).

2.3.3. Tabular data with mixed categorical and continuous features

For categorical features we can assume that the generative distribution is a categorical distribution with K categories. Then, the first integral in Eq. (5) becomes:

$$\begin{aligned}&H_\beta^{(i)}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{z}^{(j)})) = \\ &-\frac{\beta+1}{\beta}\int\delta(\mathbf{X}-\mathbf{x}^{(i)})(p_\theta(\mathbf{X}|\mathbf{z}^{(j)})^\beta - 1)d\mathbf{X} \\ &= p_\theta(\mathbf{x}_i|\mathbf{z}^{(j)})^\beta - 1\end{aligned} \tag{13}$$

The second integral can be written as:

$$\begin{aligned}&\int p_\theta(\mathbf{X}|\mathbf{z}^{(j)})^{\beta+1}d\mathbf{X} = \\ &\int\prod_{k=1}^{K}p_\theta(\mathbf{X}=k \mid \mathbf{z}^{(j)})^{\beta+1}\delta(\mathbf{X}, k)d\mathbf{X} \\ &= \sum_{k=1}^{K}p_\theta(\mathbf{X}=k \mid \mathbf{z}^{(j)})^{\beta+1}\end{aligned} \tag{14}$$

where $p_\theta(\mathbf{X}|\mathbf{z}^{(j)})^\beta = \prod p^{(j)}[\mathbf{X}=i]^\beta$

We can use the formulation derived in Section 2.3.2 for continuous variables with the assumption of Gaussian distribution. The total loss for the mixed type data then can be computed as a summation of loss from categorical and continuous features.

2.3.4. Influence function analysis

The influence function (IF) measures the effect of an abnormal observation on the training of the model. Futami et al. 2017 [25] give general expressions for IFs for both original and beta-variational inferences. Here, we analyze the IFs for Bernoulli and Gaussian cases. By studying the supremum of IFs over perturbation, we can compare the robustness of the models. The IFs for the expressions in Eqs. (3) and (4) differ in their first terms, so it is sufficient to compare the suprema of these terms only, for the non-robust case:

$$\frac{\partial}{\partial\theta}D_{KL}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z})) \tag{15}$$

and for the robust-case:

$$\frac{\partial}{\partial\theta}D_{\beta}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z})). \tag{16}$$

The IF of β-ELBO can be written as:

$$\frac{\partial}{\partial\theta}D_{\beta}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z})) \propto \sum_{i=1}^{N}\frac{\partial}{\partial\theta}p_\theta^{\beta}(\mathbf{x}_i|\mathbf{Z}). \tag{17}$$

The IF for the original ELBO is

$$\frac{\partial}{\partial\theta}D_{KL}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z})) \propto \sum_{i=1}^{N}\frac{\partial}{\partial\theta}\log p_\theta(\mathbf{x}_i|\mathbf{Z}). \tag{18}$$

Both expressions approach infinity when $p_\theta(\mathbf{x}_i|\mathbf{Z})$ goes to 0 (outliers have small probabilities) but IF for β-ELBO is upper bounded by that of ELBO when $\beta > 0$ indicating β-ELBO is less affected by outliers. As a result, RVAE will be more robust than VAE to outliers.

We further study unbounded input for the Gaussian case. Assuming $d = 1$, $\sigma = 0.5$ and defining $\hat{\mathbf{x}} = f_\theta(\mathbf{x})$ where f is a neural network parameterized by θ, The IF for our approach can be written as:

$$\frac{\partial}{\partial\theta}D_{\beta}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z})) \propto \sum_{i=1}^{N}\frac{\partial}{\partial\theta}\exp\left(-\frac{1}{2}\|f_\theta(\mathbf{x}_i)-\mathbf{x}_i\|^2\right) \tag{19}$$

which approaches 0 as $\mathbf{x}_i$ approaches $\pm$ infinity given the output of the network and the gradient are bounded. Similarly, the expression for KL divergence is:

$$\frac{\partial}{\partial\theta}D_{KL}(\hat{p}(\mathbf{X}), p_\theta(\mathbf{X}|\mathbf{Z})) \propto \sum_{i=1}^{N}\frac{\partial}{\partial\theta}\frac{1}{2}\|\hat{\mathbf{x}}_i-\mathbf{x}_i\|^2 \tag{20}$$

which is not bounded when $\mathbf{x}_i$ approaches $\pm$ infinity. Thus, β-ELBO but not ELBO is bounded for the Gaussian case.

For the Bernoulli case, we assume the sigmoid function is used as an activation function at the output layer. We can express the posterior as $p_\theta(\mathbf{x}_i|\mathbf{Z}) = f_\theta(\mathbf{x}_i)^{\mathbf{x}_i}(1-f_\theta(\mathbf{x}_i))^{1-\mathbf{x}_i}$ where $f_\theta(\mathbf{x}_i) = \frac{1}{e^{-g_\theta(\mathbf{x}_i)}}$ and $g_\theta(\mathbf{x}_i)$ is the input to the sigmoid function [25]. Then the derivative of the logarithm of the posterior in Eq. (18) with respect to the model parameters can be written as

$$\frac{\partial}{\partial\theta}\log p_\theta(\mathbf{x}_i|\mathbf{Z}) = -\mathbf{x}_i(1-f)\frac{\partial g}{\partial\theta} + (1-\mathbf{x}_i)f\frac{\partial g}{\partial\theta}. \tag{21}$$

Let us consider the case where $\mathbf{x}_i = 1$, then we have :

$$\frac{\partial}{\partial\theta}\log p_\theta(\mathbf{x}_i = 1|\mathbf{Z}) = \frac{1}{1+e^{g_\theta(\mathbf{x}_i)}}\frac{\partial g}{\partial\theta} \tag{22}$$

And for the β-ELBO, using Eq. (17) we have:

$$\frac{\partial}{\partial\theta}p_\theta^{\beta}(\mathbf{x}_i|\mathbf{Z}) = p_\theta(\mathbf{x}_i = 1|\mathbf{Z})^{\beta}\frac{\partial}{\partial\theta}\log p_\theta(\mathbf{x}_i = 1|\mathbf{Z}) = \tag{23}$$

$$\frac{1}{(1+e^{-g_\theta(\mathbf{x}_i)})^{\beta}}\frac{1}{1+e^{g_\theta(\mathbf{x}_i)}}\frac{\partial g}{\partial\theta} \tag{24}$$

Comparing these terms which are the difference between original and beta-variational inferences, stated in Eqs. (17) and (18), we can infer that $p_\theta(\mathbf{x}_i = 1|\mathbf{Z})^{\beta}$ acts as a weight on the gradient. This value is higher for normal data samples, since we have $0 < \beta$ and $0 < p_\theta(\mathbf{x}_i = 1|\mathbf{Z}) < 1$, so their gradients are weighted more highly, and have more impact on the model parameters.

3. Experimental results

We present formulations of RVAE for Gaussian and Bernoulli models as well as categorical and mixed type data and designed series of experiments to show the effectiveness RVAE for these three different choices of posterior distribution.

In each case, we optimize ELBO and β-ELBO using stochastic gradient descent with reparameterization [5]. We note that all quantitative performance evaluation below was performed on independent hold-out data.

Here we evaluate the performance of RVAE using datasets contaminated with outliers and compare it with the traditional VAE. We conducted four experiments using the MNIST [30], the EMNIST [31], the Fashion-MNIST benchmark datasets [32], two real-world Magnetic Resonance (MR) brain imaging datasets: Maryland MagNeTs study of neurotrauma (https://fitbir.nih.gov) and the Ischemic Stroke Lesion Segmentation (ISLES) database [33] (http://www.isles-challenge.org) and three benchmark datasets made available by the cyber security community: KDDCup 99 [34], NSL-KDD [35] and UNSW-NB15 [36]. Both brain imaging datasets consist of three sets of coregistered MR images corresponding to FLAIR and T1 and T2 weighting [37]. The experiments are summarized as follows:

- *EXPERIMENT 1*: in Section 3.1, using a simple simulation using MNIST as inliers as Gaussian noise as outliers to show how the encoding of VAE gets corrupted in the presence of outliers and how we can fix it using RVAE.
- *EXPERIMENT 2*: in Section 3.2, we did an outlier detection experiment using benchmark data-sets for different choices of posterior distribution Gaussian (Fashion-MNIST dataset, Shoes: inliers, other-categories: outliers) and Bernoulli models (binarized-MNIST: inliers, binarized-EMNIST: outliers. In Section 3.3, we introduced two methods for choosing the robustness parameter β and repeated experiment 2.
- *EXPERIMENT 3*: in Section 3.4, we used the Gaussian formulation for a real-word lesion detection task. Two real-world Magnetic Resonance (MR) brain imaging datasets: Maryland MagNeTs study of neurotrauma (https://fitbir.nih.gov) and the Ischemic Stroke Lesion Segmentation (ISLES) database [33] has been used.
- *EXPERIMENT 4*: in Section 3.5, we used the formulation for tabular data with mixed categorical and continuous features and applied it for a outlier detection task on three benchmark datasets made available by the cyber security community: KDDCup 99 [34], NSL-KDD [35] and UNSW-NB15 [36].

In Section 3.6, we compared RVAE performance with other methods VAE, Denoising VAE (DVAE) [38], robust AE (RAE) [20], and Coupled-VAE (CVAE) [22]. Finally in Section 3.7, we investigate the performance of VAE when the outliers in training and test data are qualitatively different we used Fashion-MNIST dataset and brain imaging datasets and Gaussian formulation.

The network architectures for VAEs were chosen based on previously established designs and summarized as below: *EXPERIMENT 1*: We use fully-connected layers with single hidden layers consisting of 400 units both for the encoder and the decoder and a bottleneck with dimension 2. *EXPERIMENT 2*: We use fully-connected layers with single hidden layers consisting of 400 units

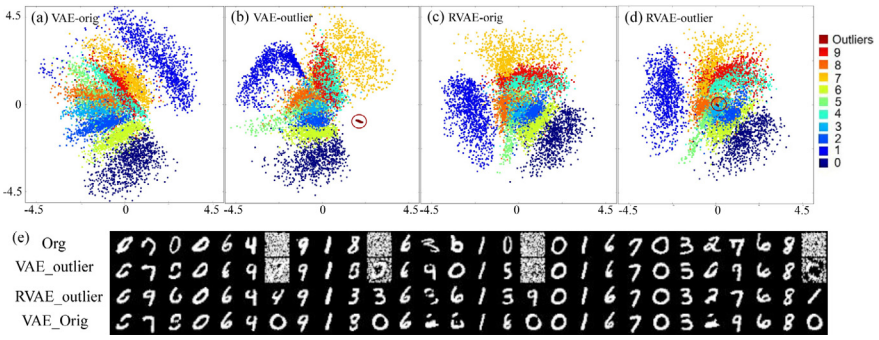

Fig. 2. Comparing robustness of VAE and RVAE using the MNIST dataset contaminated with synthetic outliers generated by Gaussian noise: (a) the 2D latent space of VAE for the original MNIST dataset without outliers (colors represent class labels of MNIST); (b) the 2D latent space of VAE for the MNIST dataset with added outliers (marked by a dark red circle); (c) the 2D latent space of RVAE without outliers; (d) the 2D latent space of RVAE with outliers added to the input data. (e) Examples of the reconstructed images using VAE (VAE_outlier: VAE trained on a data set which includes outliers, VAE_Orig: VAE trained on the outlier-free data set) and RVAE (RVAE_outlier: RVAE trained on a data set which includes outliers). Unlike the VAE, the RVAE is minimally affected by the presence of outliers in the training and reconstructs the outliers as a digit from the (outlier-free) training set.

both for the encoder and the decoder a bottleneck with dimension 20 described in [39]. *EXPERIMENT 3*: The VAE architecture consists of three consecutive blocks of convolutional layers, a batch normalization layer, a rectified linear unit (ReLU) activation function and two fully-connected layers in the bottleneck for the encoder. Similarly, the decoder consists of a fully-connected layer and three consecutive blocks of deconvolutional layers, a batch normalization layer and ReLU, with a final deconvolutional layer. *EXPERIMENT 4*: For the cyber security datasets, we use fully-connected neural networks with three hidden layers in both encoder and decoder with 128 units and tanh and softmax activation functions for continuous and categorical variables, respectively.

We used PyTorch [40] scikit-learn [41], and NumPy [42] for the implementation. We used the Adam optimizer [43] with a learning rate of 0.001 for training. For the first three experiments, bias correction parameters for the Adam optimizer were 0.9 and 0.999 (default parameters) for gradients and squared gradients, respectively. For the tabular data, these values were 0.5 and 0.999, respectively. We provide a public version of our code at https://github.com/HaleAkrami/RVAE.

3.1. Experiment 1: Effect on latent representation

First, we used the MNIST dataset comprising 70,000 28×28 grayscale images of handwritten digits [30]. We replaced 10% of the MNIST data with synthetic outlier images generated by white Gaussian noise. We binarized the data by thresholding at 0.5 of the maximum intensity value, and used the Bernoulli model of the β-ELBO (Eq. (10)) with $\beta = 0.005$. The latent dimension was chosen to be 2 by visual inspection. Fig. 2 (e) shows examples of the reconstructed images using the VAE (second row) and RVAE (third row) along with original images (first row). The results show that outlier images are encoded when VAE is used. On the other hand, RVAE, as desired, did not accurately encode the outlier noise images but rather encoded them such that they produce images consistent with the MNIST (inlier) training data after decoding. Moreover, we visually inspected the embeddings using both VAE and RVAE (Fig. 2 (a)–(d)). In the VAE case (Fig. 2 (a) and (b)), the distributions of the digits were strongly perturbed by the outlier noise images. In contrast, RVAE was not significantly affected by outliers (Fig. 2 (c) and (d)), illustrating the robustness of RVAE. For a quantitative comparison, we calculated negative BCE (Binary cross entropy) for each sample which is equivalent to the log-likelihood since we assumed a Bernoulli posterior. The average log-likelihood of outliers was much lower for the RVAE (-4036.42) than for the VAE (-545.46), while inliers had similar log-likelihood for VAE (-144.28) and RVAE (-145.97). The much larger difference between average log-likelihood of inliers and outliers for the RVAE than VAE indicates a superior ability to distinguish between the two and hence increased robustness.

3.2. Experiment 2: Reconstruction and outlier detection

For this experiment, instead of using Gaussian random noise as outliers, we replaced a fraction of the MNIST data with Extended MNIST (EMNIST) data [31] which contains images that are the same size as MNIST but do not display integers. We again binarized the data by thresholding at 0.5 of the maximum intensity value and used the Bernoulli model of the β-ELBO (Eq. (10)) for the RVAE loss function.

Similarly, we repeated the above experiment using the Fashion-MNIST dataset [32] that consists of 70,000 28×28 grayscale images of fashion products from 10 categories (7000 images per category). Here we chose shoes and sneakers as inliers classes and samples from other categories as outliers. Since these images contain a significant range of gray scales, we chose the Gaussian model for the β-ELBO (Eq. (12)). An apparently common, yet theoretically unclear practice is to use a Bernoulli model for grayscale data. This is pervasive in VAE tutorials, research literature, and default implementations of VAE in deep learning frameworks where researchers effectively treat the data as probability values rather than samples from the distribution. Using the Bernoulli model for continuous data on [0, 1] is inconsistent with the interpretation of the VAE in terms of probabilistic inference. This issue is discussed in detail in [44]. In particular, they note that even treating the algebraic form of the Bernoulli distribution as representing a continuous variable on [0, 1], the formulation is still incorrect since the distribution is not correctly normalized. Despite these theoretical concerns, to be consistent with common practice we do include results here for the Fashion-MNIST grayscale images using the Bernoulli model of the β-ELBO (Eq. (10)) with gray scale values interpreted as probabilities.

To investigate the performances of the autoencoders, we start with a fixed fraction of outliers (10%). For the MNIST-EMNIST experiment, we trained both VAE and RVAE with β varying from 0.001 to 0.02. Fig. 3 (a) shows the reconstructed images from RVAE with $\beta = 0.005, 0.01$ and 0.015 in comparison to the regular VAE. Similarly to experiment 1, with an appropriate β ($\beta = 0.01$ in this case), RVAE did not reconstruct the outliers (letters). As expected, RVAE with too small β has similar performance to the regular VAE, while RVAE with too large β rejects outliers but also rejects some normal samples.

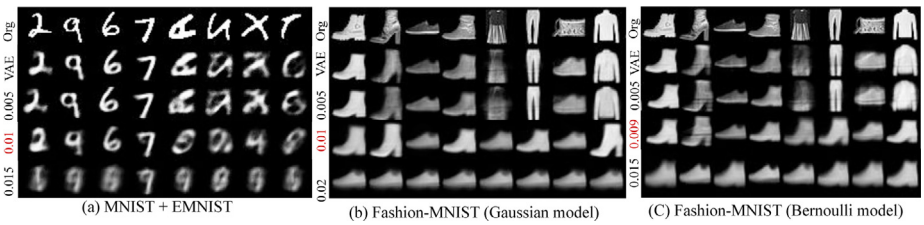

Fig. 3. Examples of reconstructed inlier images (first 4 columns in each figure) and outliers (last 4 columns in each figure) using VAE and RVAE with different βs on (a) MNIST (inliers) + EMNIST (outliers) datasets and (b,c) Images from the class of shoes (inliers) and images from the class of other accessories (outliers) in the Fashion MNIST datasets. The optimal value of β is highlighted in red.

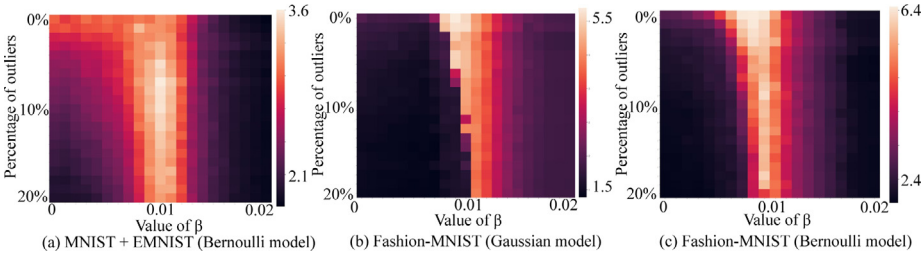

Fig. 4. The performance measure (the ratio between the overall absolute reconstruction error in outlier samples and their counterparts in the normal samples) as a function of the parameter β (x-axis) and the fraction of outliers present in the training data (y-axis) for two datasets used for experiment 2.

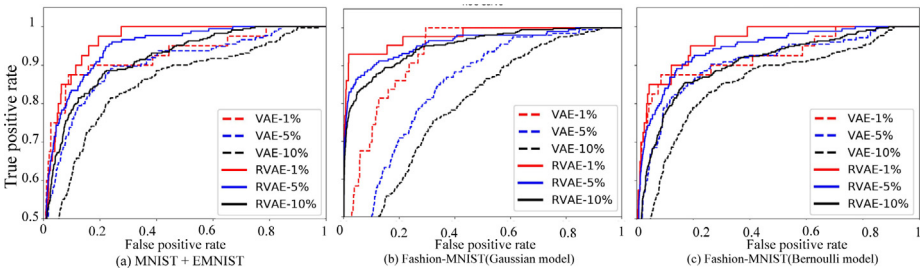

Fig. 5. ROC curves showing the performance of outlier detection using VAE and RVAE with different fractions of outliers present in the training data for the two datasets used in Experiment 2.

Next, we explored the impact of the parameter β and the fraction of outliers in the data on the performance of the RVAE. Performance was measured as the ratio between the overall absolute reconstruction error in outlier samples (letters) and their counterparts in the inlier samples (digits). The higher this metric, the more robust the model, since a robust model should in this example encode digits well but letters (outliers) poorly. Fig. 4 (a) shows the performance of this measure in a heatmap as a function of β (x-axis) and the fraction of outliers (y-axis). When only a few outliers are present, a wide range of βs (< 0.01) works almost equally well. On the other hand, when a significant fraction of the data is outliers, the best performance was achieved only when β is close to 0.01. When $\beta > 0.01$, the performance degraded regardless of the fraction of the outliers. These results are consistent with the results in Fig. 3.

We further investigate the performance of RVAE as a method for outlier detection as follows. We threshold the mean squared error between the reconstructed images and the original images. Errors exceeding a given threshold identified the image as an outlier. The resulting labels were compared to the ground truth to determine true and false-positive rates. We varied the threshold to compute Receiver Operating Characteristic (ROC) curves. Fig. 5 shows the ROC curves with RVAE shown as a solid and VAE a dashed line. The results were similar for the Fashion-MNIST dataset (Figs. 3–5 (b),(c)). The RVAE outperformed the VAE for all settings with the difference increasing with the fraction of outliers.

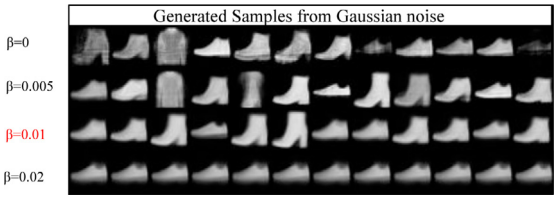

Fig. 6. Generating samples using the decoder with different values of β for the Fashion-MNIST experiment. The optimal value of β is 0.01 which generates different types of shoes. This optimal value also matches the maximum value achieved in the heat map in Fig. 4-b.

To illustrate robustness of the RVAE as a generative model we input Gaussian noise to the decoder for the Fashion-MNIST trained with 10% outliers for a range of β values. Fig. 6 shows that for $\beta = 0.01$, the network exhibits the best trade-off between generating a range of shoe images consistent with the inlier training data and non generating outliers. In contrast, smaller values ($\beta = 0, 0.005$) result in outliers affecting training, while the larger value ($\beta = 0.02$) produces almost no variability. Note that the maximum value of the heatmap in Fig. 4(b) is achieved at this optimal value of $\beta = 0.01$ when the percentage of outliers is 10%.

3.3. How to choose robustness parameter β?

In practice, outliers in the training data would not be known in advance, hence we cannot compute the reconstruction error in outlier samples in the training data as described above. Here we propose two approaches to tuning β: a semi-supervised validation based approach, and an unsupervised clustering-based approach. We used two gradient-free methods for parameter optimization: Brent's method [45,46] and Bayesian optimization [47,48].

Bayesian optimization is useful when evaluating the objective function is expensive. This approach keeps track of past evaluation results, using them to form a probabilistic model by mapping hyperparameters to the probability of a score on the objective function. This is then used to predict the next value [47,48]. Brent's method [45,46] has lower computation cost than Bayesian optimization. Brent's method involves iterative optimization using a combination of the bisection method, the secant method, and inverse quadratic interpolation. At each iteration, Brent's method estimates an optimum value of β and trains the model with that β using the training set, computes the defined metric, and based on this, finds a new estimate of β.

3.3.1. Validation-based approach

We choose a small subset of training data as a validation dataset in which we labeled inliers/outliers. Specifically, for the above MNIST-EMNIST and Fashion-MNIST experiment, we chose 1000 samples of which 10% were identified as outliers.

We trained the RVAE on the rest of the training data, and then computed the ratio of the reconstruction error for inlier samples (digits), and the outlier samples (letters) on the labeled validation dataset (Section 3.2). The lower this metric, the more robust the model is. In this example, a robust model should reconstruct digits/shoes (inliers) well but letters/other categories (outliers) poorly. We minimize this ratio with respect to β using both Brent's method and Bayesian optimization.

We demonstrate Brent's method for finding an optimal β for the MNIST-EMNIST and Fashion-MNIST experiments (Section 3.2) in Fig. 7. The red curve shows the convergence of β values. It can be seen that after only a few iterations, we are able to compute the optimal value of β. A similar optimal value (0.01 for both MNIST-EMNIST and Fashion-MNIST experiments) of β was achieved with Bayesian optimization using a maximum of 20 iterations and a log-uniform search space.

The validation-based approach needs a small validation data with samples labeled as inliers and outliers. As an alternative, for the case where such a validation set is not available, we propose a clustering-based approach as explained below.

3.3.2. Clustering-based approach

In some datasets, labeled data may not be available to generate a validation set. For such cases, we suggest a clustering-based approach. From the heatmaps in Section 3.2, it can be inferred that when the β value is too low, the reconstruction error is low both for the outliers and the inliers. Hence, they are not *clusterable* based on this measurement. Conversely, when β is large, the reconstruction error is large for both groups, so again they cannot be partitioned into separate clusters. For an optimal β value the reconstruction error should most easily allow differentiation and hence clustering of inliers and outliers into two groups. Based on this observation, we maximize the Silhouette score [49], a measure of how similar a data sample is to its cluster (cohesion) compared to other clusters (separation). The Silhouette Score is calculated using the mean intra-cluster distance x and the mean nearest-cluster distance y. The Silhouette Score for a sample is $(y - x)/\max(y, x)$. As noted above, suboptimal values of β will produce similarly small (β too small) or large (β too large) reconstruction errors so that inliers and outliers will not cluster into distinct groups using their reconstruction errors and the Silhouette score will be low. But for two clusters where outliers have large and inliers low reconstruction errors, the silhouette score will be large. Simple k-means clustering ($k = 2$) can be used to evaluate the score for each candidate value of β.

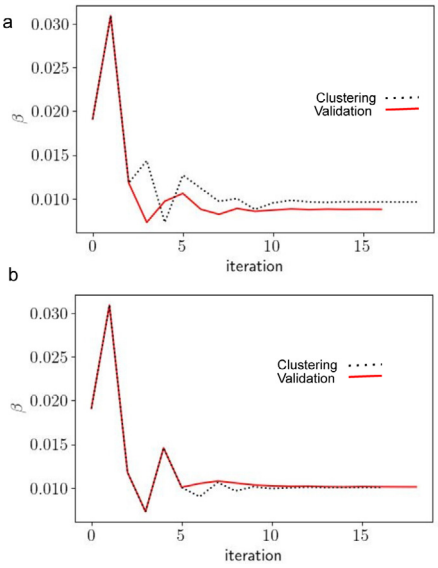

Fig. 7. Searching for the optimal β by optimizing using different metrics using Brent's method: (a) MNIST-EMNIST experiment; (b) Fashion-MNIST experiment with 10% outliers. Validation method: the ratio of reconstruction error in inliers vs outliers used as the cost function. Clustering method: the Silhouette Score for k-means cluster ($k = 2$) was used as the cost function.

The black curves in Fig. 7 show the convergence of β values using the Silhouette objective function. After only a few iterations we were able to find the optimal value of β using Brent's method. The optimal value of β using Bayesian optimization was similar (0.01) using a maximum of 20 iterations and a log-uniform search space.

3.4. Experiment 3: Detecting abnormalities in brain images using RVAE

Recently machine learning methods have been introduced to accelerate the identification of abnormal structures in medical images of the brain and other organs [12]. Since supervised methods require a large amount of annotated data, unsupervised methods have attracted considerable attention for lesion detection in brain images. A popular approach among these methods leverages VAE for anomaly detection [50] by training the VAE using nominally normal (anomaly free) data. However, if outliers, lesions or dropouts are present in the training data, VAEs cannot distinguish between normal brain images and those with outliers. Here, we tackle this real-world problem by investigating the effectiveness of the RVAE for automated detection of outliers using both simulated and real outliers. We used the VAE architecture proposed in [39] and 20 central axial slices of brain MRI datasets from 119 subjects from the Maryland MagNeTs study of neurotrauma (https://fitbir.nih.gov). We split this dataset into 107 subjects for training and 12 subjects for testing. The experiment using simulated outliers consisting of 10% of two types of outliers: random data dropout (lower intensity lines with a thickness of 5 pixels), and randomly generated simulated lesions (higher intensity Gaussian blobs). For the experiment with real outliers

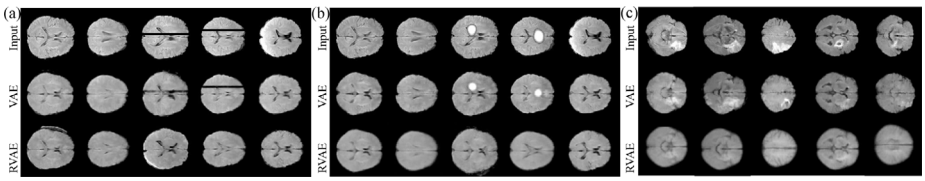

Fig. 8. Reconstructions of brain images using VAE and RVAE: (a) after randomly dropping rows with a height of 5 pixels for 10% of the Maryland MagNeTs dataset; (b) after adding simulated lesions to 10% of the Maryland MagNeTs dataset; (c) for the ISLES data with true lesions.

(lesions), we used 142 central axial slices of 24 subjects from the ISLES (The Ischemic Stroke Lesion Segmentation) database [33]. We used 21 subjects for training and 3 subjects for testing. In experiments both with simulated and real outliers, unlike VAE, RVAE is robust to the outliers in the sense that they are not reconstructed in the decoded images and can, therefore, be detected by differencing from the original images (Fig. 8). Due to the small number of samples and more variability and noise in the data, the quality of the reconstructions of examples from the dataset with real outliers is worse than that for dataset with simulated outliers. For quantitative comparison, we apply a pixel-wise ROC study for the data contaminated with simulated outliers and for the ISLES dataset. In this study each pixel is classified as either 'normal' or 'lesion' based on the thresholded error and compared to ground truth delineations of the lesions to compute true and false positive rates. By varying the threshold we were able to generate a set of ROC curves. The area under the ROC curve was 0.20 for VAE and 0.85 for RVAE for the simulated data and was 0.92 for VAE and 0.98 for RVAE for the ISLES dataset which quantitatively demonstrates the success of RVAE compared to the VAE for lesion detection.

3.5. Experiment 4: RVAE for tabular data

We compared the performance between regular VAE and RVAE by gradually contaminating the training dataset with more outliers to evaluate robustness. We use three benchmark datasets made available by the cyber security community: KDDCup 99 [34], NSL-KDD [35] and UNSW-NB15 [36]. The goal is to detect cyber attacks at the network level. All datasets are in tabular format with categorical and continuous columns. We measured the area under the receiver operating characteristic curve (AUC) as an evaluation metric.

KDDCup 99: [34] is the dataset used for "The Third Knowledge Discovery and Data Mining Tools" competition. The task was to build an automated network intrusion detector that can distinguish between attacks and normal connections. There are 41 columns of which 8 are categorical. We use the complementary 10% data for training and the labeled test data for testing.

NSL-KDD: [35] is the refined version of KDDCup 99 to resolve some of its inherent problems. Specifically, redundant connection records were removed to prevent detection models becoming biased towards frequent connection records. We used the available full training dataset for training and test dataset for testing.

UNSW-NB15: [36] This dataset was introduced by a cyber security research team from the Australian Centre for Cyber Security. We used the available partitioned datasets for training and testing. The data has 43 columns out of which, 9 features are categorical.

Thanks to the abundance of labeled data for this application, model selection for β and the early stopping was done based on the best AUC from the hold-out validation dataset (20% of the training dataset). We ran each experiment with five different initializations and report the average and standard error of AUCs across these five runs.

Table 1
Comparison of different autoencoders for MNIST+EMNIST and Fashion-MNIST anomaly detection experiment with 10% outliers and Comparison of different autoencoders for Lesion detection experiment for ISLES dataset in terms of AUC.

Dataset	VAE	RVAE	RAE	CVAE	DVAE
MNIST+EMNIST	0.84	0.90	0.89	0.86	0.84
FashionMNIST	0.79	0.96	0.93	0.82	0.79
ISLES	0.92	0.98	0.98	0.96	0.89

The results in Fig. 9 show that the performance of the VAE degrades significantly even with a small amount of contamination (1%) for all three data sets (Fig. 9 (a) KDDCup99, (b) NSLKDD, and (c) UNSW-NB15). The RVAE, on the other hand, stays robust to the outliers in the training datasets.

3.6. Comparison with other methods

We compared the performance of RVAE for outlier detection with Denoising VAE (DVAE) [38], robust AE (RAE) [20], and Coupled-VAE (CVAE) [22] for the following data sets: (i) MNIST+EMNIST (Section 3.2), (ii) a Fashion-MNIST (Section 3.2) experiment with 10% percent outliers, and (iii) a lesion detection experiment for the ISLES dataset (Section 3.4). The areas under the ROC curve (AUCs) for outlier detection are shown in Table 1. The RVAE has the highest ROC value among the methods tested. Although the RAE [20] shows competitive results and is robust to outliers it is not a generative model. Further, RAEs are only applicable for the Gaussian posterior with real-valued input data where the loss is calculated using reconstruction error. RAEs are not generally applicable, for example with categorical or tabular datasets. Furthermore, in earlier medical imaging applications, VAEs were typically shown to outperform AEs [12]. The RAE performs a decomposition of input data $\mathbf{X}$ into two components, $\mathbf{X} = \mathbf{LD} + \mathbf{S}$, where LD is a low-rank component which we want to reconstruct, and S represents a sparse component assumed to contain outliers or noise. To train the model a two-phase optimization framework is needed so that RAE is computationally more expensive than VAE. Finally, another drawback of this method is that, in contrast to the RVAE, no principled way of choosing the hyperparameter has been described.

Using the cross-entropy cost, CVAE [22] models pixel data as Bernoulli even when the data is continuous, which causes pervasive errors [44] as described in Section 3.2. Further, CVAE does neither include any general settings for other priors such as Gaussian nor provide a mechanism for tuning hyperparameters.

An alternative robust framework was proposed for the VAE using a two component mixture model for each feature in [17], where one component represents the clean data and the other robustifies the model by isolating outliers. However, that work focuses on categorical data rather than images, which is the primary focus of the current paper.

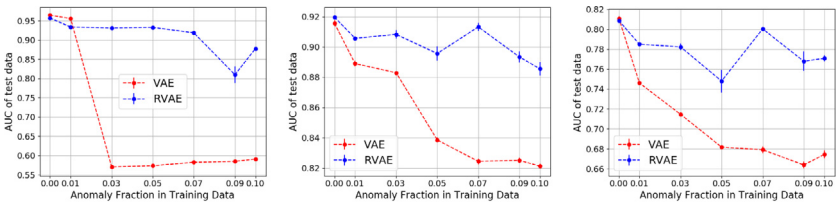

Fig. 9. Performance comparison of VAE and RVAE as function of contamination in training data for datasets: (a) KDDCup99, (b) NSLKDD, and (c) UNSW-NB15.

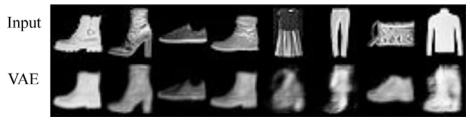

Fig. 10. Examples of reconstructed inlier images (first 4 columns in each Figure) and outliers (last 4 columns in each Figure) from the test set using original VAE when outliers in the training data are qualitatively different from those in the test data.

3.7. Inconsistency in outliers between training and testing

UPto this point we have focused on the case where training data is polluted with outliers similar to those we intend to detect. In this section we investigate the performance of VAE when the outliers in training and test data are qualitatively different.

3.7.1. Fashion-MNIST experiment

Here we added 10% outliers to both the training and test datasets. Inliers are different types of shoes and sneakers. The outliers in the training set are from EMNIST while the outliers in the test set are other fashion categories from Fashion-MNIST (the test set is similar to experiment 2). Fig. 10 shows that the standard VAE does not reconstruct the test outliers properly, as a result reconstruction error can be used to detect outliers. Consequently, in this case there may be no need to use a robust formulation. There was no significant difference in AUC (0.99) for the test set using RVAE or VAE in the outlier detection task using the reconstruction error.

3.7.2. Lesion detection

We performed the lesion detection task on 20 slices each from 15 subjects from the ISLES dataset as a test set that included outliers containing real lesions. Twenty lesion-free central axial slices of brain MRI datasets from 119 subjects from the Maryland MagNeTs study of neurotrauma (https://fitbir.nih.gov) were used for training. We separately added either simulated lesions or simulated drop-outs to these data to achieve 10% outliers to generate two different corrupted data sets for training. Note that the simulated lesions are qualitatively different in shape from their real counterparts in the test data. We trained the network using the VAE architecture proposed in [39]. Results are shown in Fig. 11. We see that the RVAE continues to perform well with both types of outliers in the training data. Conversely, while the VAE does not reconstruct the true lesions in the test input data, the reconstructions of these test images do, in some instances, contain features similar to those of the outliers used in the training data. For example in the 2nd row, 4th column for training with simulated lesions and 3rd row, 5th column for training with drop-outs. This could clearly lead to errors in detection and localizing lesions since the artifacts introduced could be interpreted as lesions when computing differences from the input images. For this experiment there was no significant difference in AUC (0.95) using VAE or RVAE because errors of the type just described occur relatively infrequently. Nevertheless, the fact that the presence of outliers in the training in the VAE can lead to artifacts in images

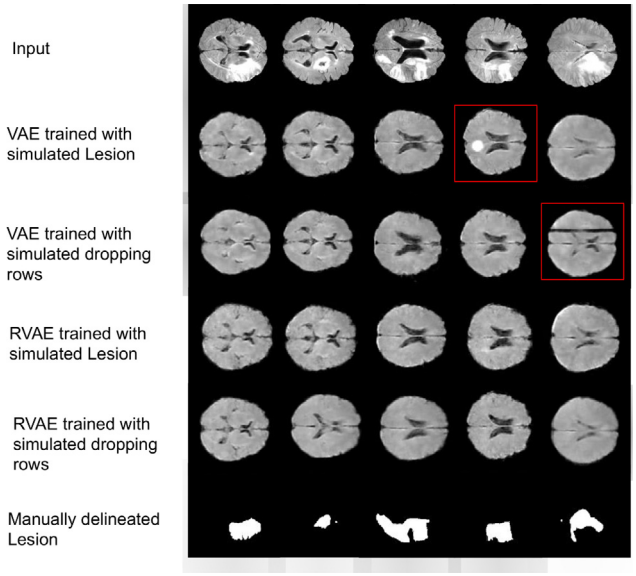

Fig. 11. Reconstructions of brain images using VAE and RVAE with inconsistency between outliers in training and testing. The VAE did not reconstruct the real lesions in the test (input) images but the reconstructions are sometimes corrupted by artifacts similar to the outliers that were present in the training data (see red squares).

as shown in the 2nd and 3rd rows of Fig. 11 argues in favor of using the RVAE over the VAE even in cases where it is suspected that outliers are different between the training and testing sets.

4. Discussion and conclusion

The presence of outliers in the form of noise, mislabeled data, and anomalies can impact the performance of machine learning models for labeling and anomaly detection tasks. In this work, we developed an effective approach for learning representations, RVAE, to ensure the robustness of learning to outliers. Our approach relies on the notion of β-divergence from robust statistics. We formulated cost functions for Bernoulli, Gaussian and categorical distributions. Furthermore, we provided an unsupervised approach to selecting the robustness hyper-parameter β in RVAE using an optimization method. We demonstrated the effectiveness of our approach using benchmark datasets from computer vision, real-world brain imaging and tabular cyber security datasets. Our experimental results indicate that the RVAE is robust to outliers in representation learning and can also be useful for outlier detection. Our approach can be used for automated anomaly detection applications in medical images and cyber security datasets.

Our results show that RVAE tends to decrease the resolution of reconstructed images relative to VAE. In our approach there is a tradeoff between robustness and quality of reconstructed images similar to the efficiency-robustness tradeoff in well-known

robust models [51]. The β divergence is an M-estimator [25] that tries to reduce the influence of outliers by applying a non-linear function on the loss; the associated efficiency-robustness tradeoff is reported in M-estimates [51]. In future work this could be addressed using an enhancement framework to increase the quality of samples generated using RVAE [52]. We note that our formulation can also be extended Generative Adversarial Networks (GANs) [53] by optimizing a divergence robust to outliers. This may also lead to improved image quality.

CRediT authorship contribution statement

Haleh Akrami: Conceptualization, Methodology, Software, Data curation, Investigation, Writing- Original draft preparation. **Anand A. Joshi:** Visualization, Software, Methodology, Writing- Reviewing and Editing. **Jian Li:** Writing- Reviewing and Editing. **Sergül Aydöre:** Methodology, Resources, Software, Writing- Reviewing and Editing. **Richard M. Leahy:** Supervision, Writing- Reviewing and Editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is supported by DOD, USA grant W81XWH-18-1-061 and also by NIH, USA grants R01 NS074980, and R01 EB026299.

Appendix. Convergence of Gaussian for $\beta = 0$

For calculating the limit at $\beta = 0$ in Eq. (12), we use L'Hôpital's rule for as follows:

$$\lim_{\beta\to 0} L_\beta(\theta,\phi;\mathbf{x}^{(i)}) = \frac{1}{(2\pi\sigma^2)^{\beta D/2}}\times$$
$$\exp\left(-\frac{\beta}{2\sigma^2}\sum_{d=1}^{D}\parallel \hat{\mathbf{x}}_d^{(j)} - \mathbf{x}_d^{(i)} \parallel^2\right) - 1 +$$
$$(\beta+1)(-\beta D/2)(2\pi\sigma^2)^{-\beta D/2-1}\times$$
$$\exp\left(-\frac{\beta}{2\sigma^2}\sum_{d=1}^{D}\parallel \hat{\mathbf{x}}_d^{(j)} - \mathbf{x}_d^{(i)} \parallel^2\right) + \tag{A.1}$$
$$\frac{1}{(2\pi\sigma^2)^{\beta D/2}}(-\frac{1}{2\sigma^2}\sum_{d=1}^{D}\|\hat{\mathbf{x}}_d^{(j)} - \mathbf{x}_d^{(i)}\|^2)\times$$
$$\exp\left(-\frac{\beta}{2\sigma^2}\sum_{d=1}^{D}\|\hat{\mathbf{x}}_d^{(j)} - \mathbf{x}_d^{(i)}\|^2\right)$$

Setting $\beta = 0$ we have:

$$\lim_{\beta\to 0} L_\beta(\theta,\phi;\mathbf{x}^{(i)}) = -\frac{1}{2\sigma^2}\sum_{d=1}^{D}\|\hat{\mathbf{x}}_d^{(j)} - \mathbf{x}_d^{(i)}\|^2 \tag{A.2}$$

assuming constant σ, $\arg\max_{\theta,\phi} L_\beta(\theta,\phi;\mathbf{x}^{(i)}) = \arg\min_{\theta,\phi} \sum_{d=1}^{D}\|\hat{\mathbf{x}}_d^{(j)} - \mathbf{x}_d^{(i)}\|^2$. Which shows the cost converges to MSE in the limiting case.

References

[1] G.E. Hinton, R.R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science 313 (5786) (2006) 504–507.
[2] U. Gather, B.K. Kale, Maximum likelihood estimation in the presence of outiliers, Comm. Statist. Theory Methods 17 (11) (1988) 3767–3784.
[3] P.J. Huber, Robust Statistics, Springer, 2011.
[4] F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, W.A. Stahel, Robust Statistics, Wiley Online Library, 1986.
[5] D.P. Kingma, M. Welling, Auto-encoding variational Bayes, 2013, arXiv preprint arXiv:1312.6114.
[6] C.M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
[7] M.J. Kusner, et al., Grammar variational autoencoder, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70, JMLR. org, 2017, pp. 1945–1954.
[8] Y. Pu, Z. Gan, R. Henao, X. Yuan, C. Li, A. Stevens, L. Carin, Variational autoencoder for deep learning of images, labels and captions, in: Advances in Neural Information Processing Systems, 2016, pp. 2352–2360.
[9] W.-N. Hsu, Y. Zhang, J. Glass, Learning latent representations for speech generation and transformation, 2017, arXiv preprint arXiv:1704.04222.
[10] J. An, S. Cho, Variational autoencoder based anomaly detection using reconstruction probability, in: Special Lecture on IE, Vol. 2, 2015, pp. 1–18.
[11] S. You, K.C. Tezcan, X. Chen, E. Konukoglu, Unsupervised lesion detection via image restoration with a normative prior, in: International Conference on Medical Imaging with Deep Learning, 2019, pp. 540–556.
[12] C. Baur, B. Wiestler, S. Albarqouni, N. Navab, Deep autoencoding models for unsupervised anomaly segmentation in brain mr images, in: International MICCAI Brainlesion Workshop, Springer, 2018, pp. 161–169.
[13] N. Pawlowski, M.C. Lee, M. Rajchl, S. McDonagh, E. Ferrante, K. Kamnitsas, S. Cooke, S. Stevenson, A. Khetani, T. Newman, et al., Unsupervised lesion detection in brain CT using bayesian convolutional autoencoders, in: MIDL, Abstract Track, Non-Archival, 2018.
[14] D. Zimmerer, S.A. Kohl, J. Petersen, F. Isensee, K.H. Maier-Hein, Context-encoding variational autoencoder for unsupervised anomaly detection, 2018, arXiv preprint arXiv:1812.05941.
[15] X. Chen, S. You, K.C. Tezcan, E. Konukoglu, Unsupervised lesion detection via image restoration with a normative prior, Med. Image Anal. (2020) 101713.
[16] E. Nalisnick, A. Matsukawa, Y.W. Teh, D. Gorur, B. Lakshminarayanan, Do deep generative models know what they don't know? 2018, arXiv preprint arXiv:1810.09136.
[17] S. Eduardo, et al., Robust variational autoencoders for outlier detection in mixed-type data, 2019, arXiv preprint arXiv:1907.06671.
[18] P. Vincent, et al., Extracting and composing robust features with denoising autoencoders, in: Proceedings of the 25th International Conference on Machine Learning, ACM, 2008, pp. 1096–1103.
[19] Y. Qi, Y. Wang, X. Zheng, Z. Wu, Robust feature learning by stacked autoencoder with maximum correntropy criterion, in: 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2014, pp. 6716–6720.
[20] C. Zhou, R.C. Paffenroth, Anomaly detection with robust deep autoencoders, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2017, pp. 665–674.
[21] Y. Zhai, B. Chen, H. Zhang, Z. Wang, Robust variational auto-encoder for radar HRRP target recognition, in: International Conference on Intelligent Science and Big Data Engineering, Springer, 2017, pp. 356–367.
[22] S. Cao, J. Li, K.P. Nelson, M.A. Kon, Coupled VAE: Improved accuracy and robustness of a variational autoencoder, 2019, arXiv preprint arXiv:1906.00536.
[23] A. Basu, I.R. Harris, N.L. Hjort, M. Jones, Robust and efficient estimation by minimising a density power divergence, Biometrika 85 (3) (1998) 549–559.
[24] S. Eguchi, S. Kato, Entropy and divergence associated with power function and the statistical application, Entropy 12 (2) (2010) 262–274.
[25] F. Futami, I. Sato, M. Sugiyama, Variational inference based on robust divergences, 2017, arXiv preprint arXiv:1710.06595.
[26] B. Dai, Y. Wang, J. Aston, G. Hua, D. Wipf, Connections with robust PCA and the role of emergent sparsity in variational autoencoder models, J. Mach. Learn. Res. 19 (1) (2018) 1573–1614.
[27] A. Zellner, Optimal information processing and Bayes's theorem, Amer. Statist. 42 (4) (1988) 278–280.
[28] A. Cichocki, S.-i. Amari, Families of alpha-beta-and gamma-divergences: Flexible and robust measures of similarities, Entropy 12 (6) (2010) 1532–1568.
[29] D. Wingate, T. Weber, Automated variational inference in probabilistic programming, 2013, arXiv preprint arXiv:1301.1299.
[30] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, et al., Gradient-based learning applied to document recognition, Proc. IEEE 86 (11) (1998) 2278–2324.
[31] G. Cohen, S. Afshar, J. Tapson, A. van Schaik, EMNIST: an extension of MNIST to handwritten letters, 2017, arXiv preprint arXiv:1702.05373.

[32] H. Xiao, K. Rasul, R. Vollgraf, Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms, 2017, arXiv preprint arXiv:1708.07747.
[33] O. Maier, et al., ISLES 2015-A public evaluation benchmark for ischemic stroke lesion segmentation from multispectral MRI, Med. Image Anal. 35 (2017) 250–269.
[34] The UCI KDD Archive, KDD cup 1999 data, 1999, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, [Online; accessed May 2020].
[35] Canadian Institute for Cybersecurity, NSL-KDD dataset, 2020, https://www.unb.ca/cic/datasets/nsl.html, [Online; accessed May 2020].
[36] ACCS, UNSW-NB15, 2020, https://www.unsw.adfa.edu.au/unsw-canberra-cyber/cybersecurity/ADFA-NB15-Datasets/, [Online; accessed May 2020].
[37] J.E. Villanueva-Meyer, M.C. Mabray, S. Cha, Current clinical brain tumor imaging, Neurosurgery 81 (3) (2017) 397–415.
[38] D. Im Im, S. Ahn, R. Memisevic, Y. Bengio, Denoising criterion for variational auto-encoding framework, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 31, 2017.
[39] A.B.L. Larsen, S.K. Sønderby, H. Larochelle, O. Winther, Autoencoding beyond pixels using a learned similarity metric, 2015, arXiv preprint arXiv:1512.09300.
[40] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch, in: NIPS-W, 2017.
[41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (Oct) (2011) 2825–2830.
[42] S.v.d. Walt, S.C. Colbert, G. Varoquaux, The NumPy array: a structure for efficient numerical computation, Comput. Sci. Eng. 13 (2) (2011) 22–30.
[43] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.
[44] G. Loaiza-Ganem, J.P. Cunningham, The continuous Bernoulli: fixing a pervasive error in variational autoencoders, in: Advances in Neural Information Processing Systems, 2019, pp. 13266–13276.
[45] R.P. Brent, Algorithms for Minimization Without Derivatives, Courier Corporation, 2013.
[46] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, et al., Numerical Recipes, Vol. 3, Cambridge University Press, Cambridge, 1989.
[47] I. Dewancker, M. McCourt, S. Clark, Bayesian optimization for machine learning: A practical guidebook, 2016, arXiv preprint arXiv:1612.04858.
[48] X. Ma, A.R. Triki, M. Berman, C. Sagonas, J. Cali, M.B. Blaschko, A Bayesian optimization framework for neural network compression, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 10274–10283.
[49] P.J. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, J. Comput. Appl. Math. 20 (1987) 53–65.
[50] X. Chen, E. Konukoglu, Unsupervised detection of lesions in brain MRI using constrained adversarial auto-encoders, 2018, arXiv preprint arXiv:1806.04972.
[51] A. Basu, S. Sarkar, The trade-off between robustness and efficiency and the effect of model smoothing in minimum disparity inference, J. Stat. Comput. Simul. 50 (3–4) (1994) 173–185.
[52] B. Dai, D. Wipf, Diagnosing and enhancing VAE models, 2019, arXiv preprint arXiv:1903.05789.
[53] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, 2014, pp. 2672–2680.