Bayesian Variational Autoencoders for Unsupervised Out-of-Distribution Detection

Erik Daxberger [1 2] **José Miguel Hernández-Lobato** [1 3 4]

Abstract

Despite their successes, deep neural networks may make unreliable predictions when faced with test data drawn from a distribution different to that of the training data, constituting a major problem for AI safety. While this has recently motivated the development of methods to detect such out-of-distribution (OoD) inputs, a robust solution is still lacking. We propose a new probabilistic, unsupervised approach to this problem based on a Bayesian variational autoencoder model, which estimates a full posterior distribution over the decoder parameters using stochastic gradient Markov chain Monte Carlo, instead of fitting a point estimate. We describe how information-theoretic measures based on this posterior can then be used to detect OoD inputs both in input space and in the model's latent space. We empirically show the effectiveness of our approach.

1. Introduction

Outlier detection in input space. While deep neural networks (DNNs) have successfully tackled complex real-world problems in various domains including vision, speech and language (LeCun et al., 2015), they still face significant limitations that make them unfit for safety-critical applications (Amodei et al., 2016). One well-known shortcoming of DNNs is when faced with test data points coming from a different distribution than the data the network saw during training, the DNN will not only output incorrect predictions, but it will do so with high confidence (Nguyen et al., 2015). The lack of robustness of DNNs to such *out-of-distribution* (OoD) inputs (or *outliers/anomalies*) was recently addressed by various methods to detect OoD inputs in the context of prediction tasks (typically classification) (Hendrycks & Gimpel, 2016; Liang et al., 2017; Hendrycks et al., 2018).

[*]Equal contribution [1]Department of Engineering, University of Cambridge [2]Empirical Inference Department, MPI for Intelligent Systems [3]Microsoft Research [4]Alan Turing Institute. Correspondence to: Erik Daxberger <ead54@cam.ac.uk>.

When we are only given input data, one simple and seemingly sensible approach to detect a potential OoD input $\mathbf{x}^*$ is to train a likelihood-based deep generative model (DGM; e.g. a VAE, auto-regressive DGM, or flow-based DGM) by (approximately) maximizing the probability $p(\mathbb{D}|\theta)$ of the training data $\mathbb{D}$ under the model parameters θ, and to then estimate the density $p(\mathbf{x}^*|\theta)$ of $\mathbf{x}^*$ under the generative model θ (Bishop, 1994). If $p(\mathbf{x}^*|\theta)$ is large, then $\mathbf{x}^*$ is likely in-distribution, and OoD otherwise. However, recent works have shown that this likelihood-based approach does not work in general, as DGMs sometimes assign *higher* density to OoD data than to in-distribution data (Nalisnick et al., 2018). While some papers developed more effective scores that correct the likelihood (Choi & Jang, 2018; Ren et al., 2019; Nalisnick et al., 2019), we argue and show that OoD detection methods which are fundamentally based on the unreliable likelihood estimates by DGMs are not robust.

Outlier detection in latent space. In a distinct line of research, recent works have tackled the challenge of optimizing a costly-to-evaluate black-box function $f : \mathbb{X} \to \mathbb{R}, f(\mathbf{x}) = y$ over a high-dimensional, richly structured input domain $\mathbb{X}$ (e.g. graphs, images). Given data $\mathbb{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$, these methods jointly train a VAE on inputs $\mathbf{x}$ and a predictive model $g : \mathbb{Z} \to \mathbb{R}, g(\mathbf{z}) = y$ mapping from latent codes $\mathbf{z}$ to targets y, to then perform the optimization w.r.t. y in the low-dimensional, continuous latent space $\mathbb{Z}$ instead of in input space $\mathbb{X}$ (Gómez-Bombarelli et al., 2018). While these methods have achieved strong results in domains including automatic chemical design and automatic machine learning (Gómez-Bombarelli et al., 2018; Luo et al., 2018; Lu et al., 2018), their practical effectiveness is limited by their ability to handle the following trade-off: They need to find inputs $\mathbf{x}$ that both have a high target value y and are sufficiently novel (i.e., not too close to training inputs $\mathbb{D}$), and at the same time ensure that the optimization w.r.t. y does not progress into regions of the latent space $\mathbb{Z}$ too far away from the training data, which might yield latent points $\mathbf{z}$ that decode to semantically meaningless or syntactically invalid inputs $\mathbf{x}$ (Kusner et al., 2017; Griffiths & Hernández-Lobato, 2017; Mahmood & Hernández-Lobato, 2019). The required ability to quantify the *novelty* of latents $\mathbf{z}$ (i.e., the semantic/syntactic distance to $\mathbb{D}$) directly corresponds to the ability to effectively detect *outliers* in latent space $\mathbb{Z}$.

Our approach. We propose a novel unsupervised, probabilistic method to *simultaneously* tackle the challenge of detecting outliers $\mathbf{x}^*$ in input space $\mathbb{X}$ as well as outliers $\mathbf{z}^*$ in latent space $\mathbb{Z}$. To this end, we take an information-theoretic perspective on OoD detection, and propose to use the (expected) *informativeness* of an input $\mathbf{x}^*$ / latent $\mathbf{z}^*$ as a proxy for whether $\mathbf{x}^*$ / $\mathbf{z}^*$ is OoD or not. To quantify this informativeness, we leverage probabilistic inference methods to maintain a posterior distribution over the parameters of a DGM, in particular of a variational autoencoder (VAE) (Kingma & Welling, 2013; Rezende et al., 2014). This results in a *Bayesian VAE* (BVAE) model, where instead of fitting a point estimate of the decoder parameters via maximum likelihood, we estimate their posterior using samples generated via stochastic gradient Markov chain Monte Carlo (MCMC). The informativeness of an unobserved input $\mathbf{x}^*$ / latent $\mathbf{z}^*$ is then quantified by measuring the (expected) change in the posterior over model parameters after having observed $\mathbf{x}^*$ / $\mathbf{z}^*$, revealing an intriguing connection to information-theoretic *active learning* (MacKay, 1992).

Our contributions. (a) We explain how DGMs can be made more robust by capturing epistemic uncertainty via a posterior distribution over their parameters, and describe how such *Bayesian DGMs* can effectively detect outliers both in input space and in the model's latent space based on information-theoretic principles (Section 3). (b) We propose a *Bayesian VAE* model as a concrete instantiation of a Bayesian DGM (Section 4). (c) We empirically demonstrate that our approach outperforms state-of-the-art OoD detection methods across several benchmarks (Section 5).

2. Problem Statement and Background

2.1. Out-of-Distribution (OoD) Detection

For *input space* OoD detection, we are given a large set $\mathbb{D} = \{\mathbf{x}_i\}_{i=1}^N$ of high-dimensional training inputs $\mathbf{x}_i \in \mathbb{X}$ (i.e., with $N > 25,000$ and $\dim(\mathbb{X}) > 500$) drawn i.i.d. from a distribution $p^*(\mathbf{x})$, and a *single* test input $\mathbf{x}^*$, and need to determine if $\mathbf{x}^*$ was drawn from p^* or from some other distribution $\tilde{p} \neq p^*$. *Latent space* OoD detection is analogous, but with an often smaller set of typically lower-dimensional latent points $\mathbf{z}_i \in \mathbb{Z}$ (i.e., with $\dim(\mathbb{Z}) < 100$).

2.2. Variational Autoencoders

Consider a latent variable model $p(\mathbf{x}, \mathbf{z}|\theta)$ with marginal log-likelihood (or *evidence*) $\log p(\mathbf{x}|\theta) = \log \int p(\mathbf{x}, \mathbf{z}|\theta) d\mathbf{z}$, where $\mathbf{x}$ are observed variables, $\mathbf{z}$ are latent variables, and θ are model parameters.[1] We assume that $p(\mathbf{x}, \mathbf{z}|\theta) = p(\mathbf{x}|\mathbf{z}, \theta)p(\mathbf{z})$ factorizes into a prior distribution $p(\mathbf{z})$ over $\mathbf{z}$ and a likelihood $p(\mathbf{x}|\mathbf{z}, \theta)$ of $\mathbf{x}$ given $\mathbf{z}$ and θ. As we assume the $\mathbf{z}$ to be continuous, $p(\mathbf{x}|\theta)$ is intractable to compute. We obtain a *variational autoencoder* (VAE) (Kingma & Welling, 2013; Rezende et al., 2014) if θ are the parameters of a DNN (the *decoder*), and the resulting intractable posterior $p(\mathbf{z}|\mathbf{x}, \theta)$ over $\mathbf{z}$ is approximated using amortized variational inference (VI) via another DNN $q(\mathbf{z}|\mathbf{x}, \phi)$ with parameters ϕ (the *encoder* or *inference/recognition network*). Given training data $\mathbb{D}$, the parameters θ and ϕ of a VAE are learned by maximizing the evidence lower bound (ELBO) $\sum_{\mathbf{x}\in\mathbb{D}} \mathcal{L}_{\theta,\phi}(\mathbf{x})$, where

$$\mathcal{L}_{\theta,\phi}(\mathbf{x}) = \mathbb{E}_{q(\mathbf{z}|\mathbf{x},\phi)}[\log p(\mathbf{x}|\mathbf{z}, \theta)] - \mathrm{KL}[q(\mathbf{z}|\mathbf{x}, \phi)\|p(\mathbf{z})] \quad (1)$$

for $\mathbf{x} \in \mathbb{D}$, with $\mathcal{L}_{\theta,\phi}(\mathbf{x}) \leq \log p(\mathbf{x}|\theta)$. As maximizing the ELBO approximately maximizes the evidence $\log p(\mathbb{D}|\theta)$, this can be viewed as approximate maximum likelihood estimation (MLE). In practice, $\mathcal{L}_{\theta,\phi}(\mathbf{x})$ in Eq. (1) is optimized by mini-batch stochastic gradient-based methods using low-variance, unbiased, stochastic Monte Carlo estimators of $\nabla\mathcal{L}_{\theta,\phi}$ obtained via the reparametrization trick. Finally, one can use *importance sampling* w.r.t. the variational posterior $q(\mathbf{z}|\mathbf{x}, \phi)$ to get an estimator $\hat{p}(\mathbf{x}|\theta, \phi)$ of the probability $p(\mathbf{x}|\theta)$ of an input $\mathbf{x}$ under the generative model, i.e.,

$$p(\mathbf{x}|\theta) \simeq \hat{p}(\mathbf{x}|\theta, \phi) = \frac{1}{K} \sum_{k=1}^{K} \frac{p(\mathbf{x}|\mathbf{z}_k, \theta)p(\mathbf{z}_k)}{q(\mathbf{z}_k|\mathbf{x}, \phi)}, \quad (2)$$

where $\mathbf{z}_k \sim q(\mathbf{z}|\mathbf{x}, \phi)$ and where the estimator $\hat{p}(\mathbf{x}|\theta, \phi)$ is conditioned on *both* θ and ϕ to make explicit the dependence on the parameters ϕ of the proposal distribution $q(\mathbf{z}|\mathbf{x}, \phi)$.

2.3. Stochastic Gradient MCMC

To generate samples $\theta \sim p(\theta|\mathbb{D})$ of parameters θ of a DNN, one typically uses stochastic gradient MCMC methods such as stochastic gradient Hamiltonian Monte Carlo (SGHMC). In particular, consider the posterior distribution $p(\theta|\mathbb{D}) \propto \exp(-U(\theta, \mathbb{D}))$ with potential energy function $U(\theta, \mathbb{D}) = -\log p(\mathbb{D}, \theta) = -\log(p(\mathbb{D}|\theta)p(\theta)) = -\sum_{\mathbf{x}\in\mathbb{D}} \log p(\mathbf{x}|\theta) - \log p(\theta)$ induced by the prior $p(\theta)$ and marginal log-likelihood $\log p(\mathbf{x}|\theta)$. Hamiltonian Monte Carlo (HMC) (Duane et al., 1987; Betancourt, 2017) is a method that generates samples $\theta \sim p(\theta|\mathbb{D})$ to efficiently explore the parameter space by simulating Hamiltonian dynamics, which involves evaluating the gradient $\nabla_\theta U(\theta)$ of U. However, computing this gradient requires examining the entire dataset $\mathbb{D}$ (due to the summation of the log-likelihood over all $\mathbf{x} \in \mathbb{D}$), which might be prohibitively costly for large datasets. To overcome this, (Chen et al., 2014) proposed SGHMC as a scalable HMC variant based on a noisy, unbiased gradient estimate $\nabla_\theta U(\theta, \mathbb{M}) \simeq \nabla_\theta U(\theta, \mathbb{D})$ computed on a minibatch $\mathbb{M}$ of points sampled uniformly at random from $\mathbb{D}$ (i.e., akin to minibatch-based optimization algorithms such as stochastic gradient descent), i.e.,

$$\nabla_\theta U(\theta, \mathbb{M}) = -\frac{|\mathbb{D}|}{|\mathbb{M}|} \sum_{\mathbf{x}\in\mathbb{M}} \nabla_\theta \log p(\mathbf{x}|\theta) - \nabla_\theta \log p(\theta). \quad (3)$$

[1] Most works include θ as a subscript, i.e., $p_{\theta_{\mathrm{MLE}}}(\mathbf{x}, \mathbf{z}) = p(\mathbf{x}, \mathbf{z}|\theta = \theta_{\mathrm{MLE}})$ with point estimate θ_{MLE}. We instead denote $p(\mathbf{x}, \mathbf{z}|\theta)$ to make clear that we assume θ to be a random variable.

3. Information-theoretic Outlier Detection

3.1. Motivation and Intuition

Why do deep generative models *fail* at OoD detection? Consider the intuitive and principled OoD detection method which first trains a density estimator parameterized by θ, and then classifies an input $\mathbf{x}^*$ as OoD based on a threshold on the density of $\mathbf{x}^*$, i.e., if $p(\mathbf{x}^*|\theta) < \tau$ (Bishop, 1994). Recent advances in deep generative modeling (DGM) allow us to do density estimation even over high-dimensional, structured input domains (e.g. images, text), which in principle enables us to use this method in such complex settings. However, the resulting OoD detection performance fundamentally relies on the quality of the likelihood estimates produced by these DGMs. In particular, a sensible density estimator should assign high density to everything within the training data distribution, and low density to everything outside – a property of crucial importance for effective OoD detection. Unfortunately, Nalisnick et al. (2018); Choi & Jang (2018) found that modern DGMs are often poorly calibrated, assigning *higher* density to OoD data than to in-distribution data.[2] This questions the use of DGMs for reliable density estimation and thus robust OoD detection.

How are deep *discriminative* models made OoD robust? DNNs are typically trained by maximizing the likelihood $p(\mathbb{D}|\theta)$ of a set of training examples $\mathbb{D}$ under model parameters θ, yielding the maximum likelihood estimate (MLE) θ^* (Goodfellow et al., 2016). For *discriminative*, predictive models $p(y|\mathbf{x}, \theta)$, it is well known that the point estimate θ^* does not capture *model / epistemic uncertainty*, i.e., uncertainty about the choice of model parameters θ induced by the fact that many different models θ might have generated $\mathbb{D}$. As a result, discriminative models $p(y|\mathbf{x}, \theta^*)$ trained via MLE tend to be overconfident in their predictions, especially on OoD data (Nguyen et al., 2015; Guo et al., 2017). A principled, established way to capture model uncertainty in DNNs is to be Bayesian and infer a full *distribution* $p(\theta|\mathbb{D})$ over parameters θ, yielding the predictive distribution $p(y|\mathbf{x}, \mathbb{D}) = \int p(y|\mathbf{x}, \theta)p(\theta|\mathbb{D})d\theta$ (Gal, 2016). Bayesian DNNs have much better OoD robustness than deterministic ones (e.g., producing low uncertainty for in-distribution and high uncertainty for OoD data), and OoD calibration has become a major benchmark for Bayesian DNNs (Gal & Ghahramani, 2016; Ovadia et al., 2019; Osawa et al., 2019; Maddox et al., 2019). This suggests that capturing model uncertainty via Bayesian inference is a promising way to achieve robust, principled OoD detection.

[2]It is a common misconception that DGMs are "immune" to OoD miscalibration as they capture a density. While this might hold for simple models such as KDEs (Parzen, 1962) on low-dimensional data, it does *not* generally hold for complex, DNN-based models on high-dimensional data (Nalisnick et al., 2018). In particular, while DGMs are trained to assign high probability to the training data, OoD data is not necessarily assigned low probability.

Why should we use *Bayesian DGMs* for OoD detection? Just like deep discriminative models, DGMs are typically trained by maximizing the probability $p(\mathbb{D}|\theta)$ that $\mathbb{D}$ was generated by the density model, yielding the MLE θ^*. As a result, it is not surprising that the shortcomings of MLE-trained *discriminative* models also translate to MLE-trained *generative* models, such as the miscalibration and unreliability of their likelihood estimates $p(\mathbf{x}^*|\theta^*)$ for OoD inputs $\mathbf{x}^*$. This is because there will always be many different plausible generative / density models θ of the training data $\mathbb{D}$, which are not captured by the point estimate θ^*. If we do not trust our predictive models $p(y|\mathbf{x}, \theta^*)$ on OoD data, why should we trust our generative models $p(\mathbf{x}|\theta^*)$, given that both are based on the same, unreliable DNNs? In analogy to the discriminative setting, we argue that OoD robustness can be achieved by capturing the epistemic uncertainty in the DGM parameters θ. This motivates the use of *Bayesian* DGMs, which estimate a full *distribution* $p(\theta|\mathbb{D})$ over parameters θ and thus capture many different density estimators to explain the data, yielding the *expected/average* likelihood

$$p(\mathbf{x}|\mathbb{D}) = \int p(\mathbf{x}|\theta)p(\theta|\mathbb{D})d\theta = \mathbb{E}_{p(\theta|\mathbb{D})}[p(\mathbf{x}|\theta)] \,. \quad (4)$$

***How* can we use Bayesian DGMs for OoD detection?** Assume that given data $\mathbb{D}$, we have inferred a distribution $p(\theta|\mathbb{D})$ over the parameters θ of a DGM. In particular, we consider the case where $p(\theta|\mathbb{D})$ is represented by a set $\{\theta_m\}_{m=1}^M$ of M *samples* $\theta_m \sim p(\theta|\mathbb{D})$, which can be viewed as an *ensemble* of M DGMs.[3] Our goal now is to decide if a given, new input $\mathbf{x}^*$ is in-distribution or OoD. To this end, we *refrain* from classifying $\mathbf{x}^*$ as OoD based on a threshold on the (miscalibrated) likelihoods that one or more of the models $\{\theta_m\}_{m=1}^M$ assign to $\mathbf{x}^*$ (Bishop, 1994).[4] **Instead, we propose to use a threshold on a measure $D[\cdot]$ of the *variation* or *disagreement* in the likelihoods $\{p(\mathbf{x}^*|\theta_m)\}_{m=1}^M$ of the different models $\{\theta_m\}_{m=1}^M$, i.e., to classify an input $\mathbf{x}^*$ as OoD if $D[\{p(\mathbf{x}^*|\theta_m)\}_{m=1}^M] < \tau$.** In particular, if the models $\{\theta_m\}_{m=1}^M$ *agree* as to how probable $\mathbf{x}^*$ is, then $\mathbf{x}^*$ likely is an *in-distribution* input. Conversely, if the models $\{\theta_m\}_{m=1}^M$ *disagree* as to how probable $\mathbf{x}^*$ is, then $\mathbf{x}^*$ likely is an *OoD* input.[5] This intuitive decision rule is a direct consequence of the property that the epistemic uncertainty of a parametric model θ (which is exactly what the variation/disagreement across models $\{\theta_m\}_{m=1}^M$ captures) is naturally low for in-distribution and high for OoD inputs – the very same property that makes Bayesian discriminative DNNs robust to OoD inputs.

[3]This setting covers both parametric family methods such as variational inference and sampling-based methods such as MCMC.

[4]Note that $p(\mathbf{x}^*|\mathbb{D}) \simeq \frac{1}{M}\sum_{m=1}^M p(\mathbf{x}^*|\theta_m)$ in Eq. (4) remains unreliable if $\mathbf{x}^*$ is OoD. E.g., Nalisnick et al. (2018) show that averaging likelihoods across an ensemble of DGMs *does not help*.

[5]If the $\{\theta_m\}_{m=1}^M$ were perfect density estimators, they would all *agree* that an OoD input $\mathbf{x}^*$ is *unlikely*. But, model uncertainty makes the DGMs *disagree* on $\{p(\mathbf{x}^*|\theta_m)\}_{m=1}^M$ if $\mathbf{x}^*$ is OoD.

3.2. Quantifying Disagreement between Models

We propose the following score $D_\Theta[\mathbf{x}^*]$ to quantify the disagreement or variation in the likelihoods $\{p(\mathbf{x}^*|\theta_m)\}_{m=1}^M$ of a set $\{\theta_m\}_{m=1}^M$ of model parameter samples $\theta_m \sim p(\theta|\mathbb{D})$:

$$D_\Theta[\mathbf{x}^*] = \frac{1}{\sum_{\theta\in\Theta} w_\theta^2}, \quad \text{with} \quad w_\theta = \frac{p(\mathbf{x}^*|\theta)}{\sum_{\theta\in\Theta} p(\mathbf{x}^*|\theta)} \,. \quad (5)$$

I.e., the likelihoods $\{p(\mathbf{x}^*|\theta_m)\}_{m=1}^M$ are first *normalized* to yield $\{w_{\theta_m}\}_{m=1}^M$ (see Eq. (5)), such that $w_\theta \in [0,1]$ and $\sum_{\theta\in\Theta} w_\theta = 1$. The normalized likelihoods $\{w_{\theta_m}\}_{m=1}^M$ effectively define a categorical distribution over models $\{\theta_m\}_{m=1}^M$, where each value w_θ can be interpreted as the probability that $\mathbf{x}^*$ was generated from the model θ, thus measuring how well $\mathbf{x}^*$ is *explained* by the model θ, relative to the other models. To obtain the score $D_\Theta[\mathbf{x}^*]$ in Eq. (5), we then square the normalized likelihoods, sum them up, and take the reciprocal. Note that $D_\Theta[\mathbf{x}^*] \in [1, M], \forall \mathbf{x}^*$.[6] For *latent points* $\mathbf{z}^* \in \mathbb{Z}$, we take into account all possible inputs $\mathbf{x}^* \in \mathbb{X}$ corresponding to $\mathbf{z}^*$, yielding the *expected* disagreement $D_\Theta[\mathbf{z}^*] = \mathbb{E}_{p(\mathbf{x}|\mathbf{z}^*)}[D_\Theta[\mathbf{x}]] \simeq \frac{1}{N}\sum_{n=1}^N D_\Theta[\mathbf{x}_n]$, with inputs $\mathbf{x}_n$ sampled from the conditional distribution $p(\mathbf{x}|\mathbf{z}^*)$, and $D_\Theta[\mathbf{x}]$ defined as in Eq. (5).

As $D_\Theta[\cdot]$ measures the degree of disagreement/variation between the models $\{\theta_m\}_{m=1}^M$ as to how probable $\mathbf{x}^*$/$\mathbf{z}^*$ is, it can be used to classify $\mathbf{x}^*$/$\mathbf{z}^*$ as follows: If $D_\Theta[\cdot]$ is large, then $[w_\theta]_{\theta\in\Theta}$ is close to the (discrete) uniform distribution $[\frac{1}{M}]_{\theta\in\Theta}$ (for which $D_\Theta[\cdot] = M$), meaning that all models $\theta \in \Theta$ explain $\mathbf{x}^*$/$\mathbf{z}^*$ equally well and are in agreement as to how probable $\mathbf{x}^*$/$\mathbf{z}^*$ is. Thus, $\mathbf{x}^*$/$\mathbf{z}^*$ likely is *in-distribution*. Conversely, if $D_\Theta[\cdot]$ is small, then $[w_\theta]_{\theta\in\Theta}$ contains a few large weights (i.e., corresponding to models that by chance happen to explain $\mathbf{x}^*$/$\mathbf{z}^*$ well), with all other weights being very small, where in the extreme case, $[w_\theta]_{\theta\in\Theta} = [0,\dots,0,1,0,\dots,0]$ (for which $D_\Theta[\cdot] = 1$). This means that the models do not agree as to how probable $\mathbf{x}^*$/$\mathbf{z}^*$ is, so that $\mathbf{x}^*$/$\mathbf{z}^*$ likely is *out-of-distribution*.

3.3. An Information-theoretic Perspective

We now provide a more principled justification for the disagreement score $D_\Theta[\mathbf{x}^*]$ in Eq. (5), which induces an *information-theoretic perspective* on OoD detection and reveals an intriguing connection to *active learning*. Assume that given training data $\mathbb{D}$ and a prior distribution $p(\theta)$ over the DGM parameters θ, we have inferred a *posterior distribution* $p(\theta|\mathbb{D}) = \frac{p(\mathbb{D}|\theta)p(\theta)}{\int p(\mathbb{D}|\theta)p(\theta)d\theta} = \frac{p(\mathbb{D}|\theta)}{p(\mathbb{D})}p(\theta)$ over θ. Then, for a given input $\mathbf{x}^*$, the score $D_\Theta[\mathbf{x}^*]$ quantifies *how much the posterior* $p(\theta|\mathbb{D})$ *would change* if we were to add $\mathbf{x}^*$ to $\mathbb{D}$ and then infer the *augmented posterior* $p(\theta|\mathbb{D}^*) = \frac{p(\mathbf{x}^*|\theta)p(\theta|\mathbb{D})}{\int p(\mathbf{x}^*|\theta)p(\theta|\mathbb{D})d\theta} = \frac{p(\mathbf{x}^*|\theta)}{p(\mathbf{x}^*|\mathbb{D})}p(\theta|\mathbb{D})$ based on this new training set $\mathbb{D}^* = \mathbb{D}\cup\{\mathbf{x}^*\}$. To see this, first note that this change in the posterior is quantified by the *normalized likelihood* $\frac{p(\mathbf{x}^*|\theta)}{p(\mathbf{x}^*|\mathbb{D})}$, such that models θ under which $\mathbf{x}^*$ is more (less) likely – relative to all other models – will have a higher (lower) probability under the updated posterior $p(\theta|\mathbb{D}^*)$. Now, given the samples $\{\theta_m\}_{m=1}^M$ of the old posterior $p(\theta|\mathbb{D})$, the normalized likelihood $\frac{p(\mathbf{x}^*|\theta)}{p(\mathbf{x}^*|\mathbb{D})}$ for a given model θ is proportional to w_θ in Eq. (5), i.e.,

$$\frac{p(\mathbf{x}^*|\theta)}{p(\mathbf{x}^*|\mathbb{D})} \overset{(4)}{=} \frac{p(\mathbf{x}^*|\theta)}{\mathbb{E}_{p(\theta|\mathbb{D})}[p(\mathbf{x}^*|\theta)]} \simeq \frac{p(\mathbf{x}^*|\theta)}{\frac{1}{M}\sum_{\theta\in\Theta} p(\mathbf{x}^*|\theta)} \overset{(5)}{=} M w_\theta \,. \quad (6)$$

Thus, w_θ intuitively measures the *relative usefulness* of θ for describing the new posterior $p(\theta|\mathbb{D}^*)$. More formally, the $[w_\theta]_{\theta\in\Theta}$ correspond to the *importance weights* of the samples $\theta \in \Theta$ drawn from the *proposal* distribution $p(\theta|\mathbb{D})$ for an importance sampling-based Monte Carlo approximation of an expectation w.r.t. the *target* distribution $p(\theta|\mathbb{D}^*)$,

$$\mathbb{E}_{p(\theta|\mathbb{D}^*)}[f(\theta)] \overset{(6)}{\simeq} \mathbb{E}_{p(\theta|\mathbb{D})}[M w_\theta f(\theta)] \simeq \sum_{\theta\in\Theta} w_\theta f(\theta) \quad (7)$$

for any function $f : \Theta \to \mathbb{R}$. The score $D_\Theta[\mathbf{x}^*]$ in Eq. (5) is a widely used measure of the efficiency of the estimator in Eq. (7), known as the *effective sample size* (ESS) of $\{\theta_m\}_{m=1}^M$ (Martino et al., 2017). It quantifies how many i.i.d. samples drawn from the target posterior $p(\theta|\mathbb{D}^*)$ are equivalent to the M samples $\theta \in \Theta$ drawn from the proposal posterior $p(\theta|\mathbb{D})$ and weighted according to w_θ, and thus indeed measures the *change in distribution* from $p(\theta|\mathbb{D})$ to $p(\theta|\mathbb{D}^*)$. Equivalently, $D_\Theta[\mathbf{x}^*]$ can be viewed as quantifying the *informativeness* of $\mathbf{x}^*$ for updating the DGM parameters θ to the ones capturing the true density.[7]

The OoD detection mechanism described in Section 3.2 can thus be intuitively summarised as follows: *In-distribution* inputs $\mathbf{x}^*$ are similar to the data points already in $\mathbb{D}$ and thus *uninformative* about the model parameters θ, inducing *small* change in distribution from $p(\theta|\mathbb{D})$ to $p(\theta|\mathbb{D}^*)$, resulting in a *large* ESS $D_\Theta[\mathbf{x}^*]$. Conversely, *OoD* inputs $\mathbf{x}^*$ are very different from the previous observations in $\mathbb{D}$ and thus *informative* about the model parameters θ, inducing *large* change in the posterior, resulting in a *small* ESS $D_\Theta[\mathbf{x}^*]$.

Finally, this information-theoretic perspective on OoD detection reveals a close relationship to *information-theoretic active learning* (MacKay, 1992; Houlsby et al., 2011). There, the same notion of *informativeness* (or, equivalently, *disagreement*) is used to quantify the *novelty* of an input $\mathbf{x}^*$ to be added to the data $\mathbb{D}$, aiming to maximally improve the estimate of the model parameters θ by maximally *reducing the entropy / epistemic uncertainty* in the posterior $p(\theta|\mathbb{D})$.[7]

[6] $D[\cdot]$ in Eq. (5) is closely related to the *sample variance* of the likelihoods, $\text{Var}_\Theta[\mathbf{x}^*] = \frac{1}{M}\sum_{m=1}^M (p(\mathbf{x}^*|\theta_m) - \mu_\Theta[\mathbf{x}^*])^2$, $\mu_\Theta[\mathbf{x}^*] = \frac{1}{M}\sum_{m=1}^M p(\mathbf{x}^*|\theta_m)$: While $D[\cdot]$ is the *reciprocal* of the second *raw* moment of the *normalized* likelihoods, the variance is the second *central* moment of the *unnormalized* likelihoods. Thus, $D_\Theta[\mathbf{x}^*] \in [1, M]$ is bounded, while $\text{Var}_\Theta[\mathbf{x}^*] \in [0, \infty)$ is unbounded. Finally, both scores yield the same ordering of inputs, i.e., $\text{Var}_\Theta[\mathbf{x}_1] > \text{Var}_\Theta[\mathbf{x}_2] \Leftrightarrow D_\Theta[\mathbf{x}_1] < D_\Theta[\mathbf{x}_2], \forall \mathbf{x}_1, \mathbf{x}_2 \in \mathbb{X}$.

[7] This connection is described in further detail in the appendix.

4. The Bayesian VAE (BVAE)

As an example of a Bayesian DGM, we propose a *Bayesian VAE* (BVAE), where instead of fitting the model parameters θ via (approximate) MLE, $\theta_{\text{MLE}} = \arg\max_\theta \mathcal{L}_{\theta,\phi}(\mathbb{D})$, to get the likelihood $p(\mathbf{x}|\mathbf{z}, \theta_{\text{MLE}})$, we place a prior $p(\theta)$ over θ and estimate its posterior $p(\theta|\mathbb{D}) \propto p(\mathbb{D}|\theta)p(\theta)$, yielding the likelihood $p(\mathbf{x}|\mathbf{z}, \mathbb{D}) = \int p(\mathbf{x}|\mathbf{z}, \theta)p(\theta|\mathbb{D})d\theta$. The *marginal likelihood* $p(\mathbf{x}|\mathbb{D}) = \int\int p(\mathbf{x}|\mathbf{z}, \theta)p(\mathbf{z})d\mathbf{z}p(\theta|\mathbb{D})d\theta$ thus integrates out *both* the latent variables $\mathbf{z}$ and model parameters θ (cf. Eq. (4)). The resulting generative process draws a $\mathbf{z} \sim p(\mathbf{z})$ from its prior and a $\theta \sim p(\theta|\mathbb{D})$ from its posterior, and then generates $\mathbf{x} \sim p(\mathbf{x}|\mathbf{z}, \theta)$ via the likelihood. Training a BVAE thus requires Bayesian inference of *both* the posterior $p(\mathbf{z}|\mathbf{x}, \mathbb{D})$ over $\mathbf{z}$ and the posterior $p(\theta|\mathbb{D})$ over θ, which is both intractable and thus requires approximation. We propose two variants for inferring the posteriors over $\mathbf{z}$ and θ in a BVAE.

4.1. Variant 1: BVAE with a Single Fixed Encoder

a) Learning the encoder parameters ϕ. As in a regular VAE, we approximate the posterior $p(\mathbf{z}|\mathbf{x}, \mathbb{D})$ using amortized VI via a recognition network $q(\mathbf{z}|\mathbf{x}, \phi)$ whose parameters ϕ are fit by maximizing the ELBO $\mathcal{L}_{\theta,\phi}(\mathbf{x})$ in Eq. (1), i.e., $\phi^* = \arg\max \mathcal{L}_{\theta,\phi}(\mathbf{x})$, yielding a single fixed encoder.

b) Learning the decoder parameters θ. To generate posterior samples $\theta \sim p(\theta|\mathbb{D})$ of decoder parameters, we propose to use SGHMC (cf. Section 2). However, the gradient of the energy function $\nabla_\theta U(\theta, \mathbb{M})$ in Eq. (3) used for simulating the Hamiltonian dynamics requires evaluating the log-likelihood $\log p(\mathbf{x}|\theta)$, which is intractable in a BVAE (as in a VAE). To alleviate this, we approximate the log-likelihood appearing in $\nabla_\theta U(\theta, \mathbb{M})$ by the ordinary VAE ELBO $\mathcal{L}_{\theta,\phi}(\mathbf{x})$ in Eq. (1). Given a set $\Theta = \{\theta_m\}_{m=1}^M$ of posterior samples $\theta_m \sim p(\theta|\mathbb{D})$, we can more intuitively think of having a finite mixture/ensemble of decoders/generative models $p(\mathbf{x}|\mathbf{z}, \mathbb{D}) = \mathbb{E}_{p(\theta|\mathbb{D})}[p(\mathbf{x}|\mathbf{z}, \theta)] \simeq \frac{1}{M}\sum_{\theta\in\Theta} p(\mathbf{x}|\mathbf{z}, \theta)$.

c) Likelihood estimation. This BVAE variant is effectively trained like a normal VAE, but using a sampler instead of an optimizer for θ (pseudocode is found in the appendix). We obtain an ensemble of M VAEs (ϕ^*, θ_m) with a single *shared* encoder ϕ^* and M separate decoder samples θ_m; see Fig. 1 (left) for a cartoon illustration. For the m-th VAE, the likelihood $p(\mathbf{x}|\theta_m) \simeq \hat{p}(\mathbf{x}|\theta_m, \phi^*)$ can then be estimated via importance sampling w.r.t. $q(\mathbf{z}|\mathbf{x}, \phi^*)$, as in Eq. (2).

4.2. Variant 2: BVAE with a *Distribution* over Encoders

a) Learning the encoder parameters ϕ. Recall that amortized VI aims to *learn* to do posterior inference, by optimizing the parameters $\phi^* = \arg\max_\phi \mathcal{L}(\mathbb{D})_{\theta,\phi}$ (cf. Eq. (1)) of an inference network $i_\phi(\mathbf{x}) = \psi$ mapping inputs $\mathbf{x}$ to parameters ψ of the variational posterior $q_\psi(\mathbf{z}) = q(\mathbf{z}|\mathbf{x}, \phi)$ over $\mathbf{z}$.

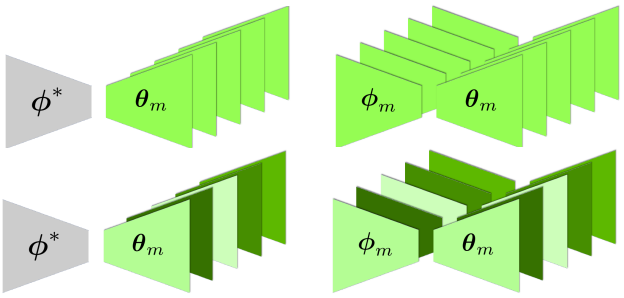

Figure 1. Illustrations of BVAEs with $M = 5$. (Left) variant 1 with shared encoder ϕ^* and M decoders and (right) variant 2 as an ensemble of M VAEs, with (top) *agreement* and (bottom) *disagreement* in likelihoods $p(\mathbf{x}^*|\theta_m)$ (as encoded by color intensity).

However, one major shortcoming of fitting a single encoder parameter setting ϕ^* is that $q(\mathbf{z}|\mathbf{x}, \phi^*)$ will *not* generalize to OoD inputs, but will instead produce confidently wrong posterior inferences (Cremer et al., 2018; Mattei & Frellsen, 2018) (cf. Section 3.1). To alleviate this, we instead capture multiple encoders by inferring a *distribution* over the variational parameters ϕ. While this might appear odd conceptually, it allows us to quantify our *epistemic uncertainty in the amortized inference* of $\mathbf{z}$. It might also be interpreted as regularizing the encoder (Shu et al., 2018), or as increasing its flexibility (Yin & Zhou, 2018). We thus also place a prior $p(\phi)$ over ϕ and infer the posterior $p(\phi|\mathbb{D})$, yielding the amortized posterior $q(\mathbf{z}|\mathbf{x}, \mathbb{D}) = \int q(\mathbf{z}|\mathbf{x}, \phi)p(\phi|\mathbb{D})d\phi$. We also use SGHMC to sample $\phi_m \sim p(\phi|\mathbb{D})$, again using the ELBO $\mathcal{L}_{\theta,\phi}(\mathbf{x})$ in Eq. (1) to compute $\nabla_\phi U(\phi, \mathbb{M})$ (cf. Eq. (3)). Given a set $\Phi = \{\phi_m\}_{m=1}^M$ of posterior samples $\phi_m \sim p(\phi|\mathbb{D})$, we can again more intuitively think of having as a finite mixture/ensemble of encoders/inference networks $q(\mathbf{z}|\mathbf{x}, \mathbb{D}) = \mathbb{E}_{p(\phi|\mathbb{D})}[q(\mathbf{z}|\mathbf{x}, \phi)] \simeq \frac{1}{M}\sum_{\phi\in\Phi} q(\mathbf{z}|\mathbf{x}, \phi)$.

b) Learning the decoder parameters θ. We sample $\theta \sim p(\theta|\mathbb{D})$ as in Section 4.1. The only difference is that we now have the encoder mixture $q(\mathbf{z}|\mathbf{x}, \mathbb{D})$ instead of the single encoder $q(\mathbf{z}|\mathbf{x}, \phi)$, technically yielding the ELBO $\mathcal{L}_\theta(\mathbf{x}) = \mathbb{E}_{q(\mathbf{z}|\mathbf{x},\mathbb{D})}[\log p(\mathbf{x}|\mathbf{z}, \theta)] - \text{KL}[q(\mathbf{z}|\mathbf{x}, \mathbb{D})\|p(\mathbf{z})]$ which depends on θ only, as ϕ is averaged over $p(\phi|\mathbb{D})$. However, in practice, we for simplicity only use the most recent sample $\phi_m \sim p(\phi|\mathbb{D})$ to estimate $q(\mathbf{z}|\mathbf{x}, \mathbb{D}) \simeq q(\mathbf{z}|\mathbf{x}, \phi_m)$, such that $\mathcal{L}_\theta(\mathbf{x})$ effectively reduces to the normal VAE ELBO in Eq. (1) with fixed encoder ϕ_m.

c) Likelihood estimation. This BVAE variant is effectively trained like a normal VAE, but using a sampler instead of an optimizer for *both* ϕ and θ (pseudocode is found in the appendix). We obtain an ensemble of M VAEs (ϕ_m, θ_m) with M pairs of coupled encoder-decoder samples; see Fig. 1 (right) for a cartoon illustration. For the m-th VAE, the likelihood $p(\mathbf{x}|\theta_m) \simeq \hat{p}(\mathbf{x}|\theta_m, \phi_m)$ can then be estimated via importance sampling w.r.t. $q(\mathbf{z}|\mathbf{x}, \phi_m)$, as in Eq. (2).

5. Experiments

5.1. Out-of-Distribution Detection in Input Space

BVAE details. We assess both proposed BVAE variants: $\texttt{BVAE}_1$ samples θ and optimizes ϕ (see Section 4.1), while $\texttt{BVAE}_2$ samples *both* θ and ϕ (see Section 4.2). Our `PyTorch` implementation uses Adam (Kingma & Ba, 2014) with learning rate 10^{-3} for optimization, and scale-adapted SGHMC with step size 10^{-3} and momentum decay 0.05 (Springenberg et al., 2016) for sampling[8]. Following Chen et al. (2014); Springenberg et al. (2016), we place Gaussian priors over θ and ϕ, i.e., $p(\theta) = \mathcal{N}(0, \lambda_\theta^{-1})$ and $p(\phi) = \mathcal{N}(0, \lambda_\phi^{-1})$, and Gamma hyperpriors over the precisions λ_θ and λ_ϕ, i.e., $p(\lambda_\theta) = \Gamma(\alpha_\theta, \beta_\theta)$ and $p(\lambda_\phi) = \Gamma(\alpha_\phi, \beta_\phi)$, with $\alpha_\theta = \beta_\theta = \alpha_\phi = \beta_\phi = 1$, and resample λ_θ and λ_ϕ after every training epoch (i.e., a full pass over $\mathbb{D}$). We discard samples within a burn-in phase of $B = 1$ epoch and store a sample after every $D = 1$ epoch, which we found to be robust and effective choices.

Experimental setup. We use three benchmarks: (a) FashionMNIST (in-distribution) vs. MNIST (OoD) (Hendrycks et al., 2018; Nalisnick et al., 2018; Zenati et al., 2018; Akcay et al., 2018; Ren et al., 2019), (b) SVHN (in-distribution) vs. CIFAR10 (OoD)[9] (Hendrycks et al., 2018; Nalisnick et al., 2018; Choi & Jang, 2018), and (c) eight classes of FashionMNIST (in-distribution) vs. the remaining two classes (OoD), using five different splits $\{(0,1),(2,3),(4,5),(6,7),(8,9)\}$ of held-out classes (Ahmed & Courville, 2019). We compare against the log-likelihood (`LL`) as well as all three state-of-the-art methods for unsupervised OoD detection described in Section 6: (1) The generative ensemble based method by Choi & Jang (2018) composed of five independently trained models (`WAIC`), (2) the likelihood ratio method by Ren et al. (2019) (`LLR`), using Bernoulli rates $\mu = 0.2$ for the FashionMNIST vs. MNIST benchmark (Ren et al., 2019), and $\mu = 0.15$ for the other benchmarks, and (3) the test for typicality by Nalisnick et al. (2019) (`TT`). All methods use VAEs for estimating log-likelihoods.[10] For evaluation, we randomly select 5000 in-distribution and OoD inputs from held-out test sets and compute the following, threshold independent metrics (Hendrycks & Gimpel, 2016; Liang et al., 2017; Hendrycks et al., 2018; Alemi et al., 2018; Ren et al., 2019): (i) The area under the ROC curve (AUROC↑), (ii) the area under the precision-recall curve (AUPRC↑), and (iii) the false-positive rate at 80% true-positive rate (FPR80↓).

[8] We use the implementation of SGHMC as a `PyTorch Optimizer` at `https://github.com/automl/pybnn`.

[9] Unlike Nalisnick et al. (2018), we found the likelihood calibration to be poor on this benchmark (Fig. 2, bottom middle, shows the overlap in likelihoods) and decent on the opposite benchmark.

[10] We use Nalisnick et al. (2018); Choi & Jang (2018); Ren et al. (2019)'s convolutional VAE architecture and training protocol.

Results. Table 1 shows that both BVAE variants *significantly outperform* the other methods on the considered benchmarks. Fig. 2 (left column) shows the ROC curves used to compute the AUROC metric in Table 1, for the FashionMNIST vs. MNIST (top) and SVHN vs. CIFAR10 (bottom) benchmarks; ROC curves for the FashionMNIST (held-out) benchmark as well as precision-recall curves for all benchmarks are found in the appendix. $\texttt{BVAE}_2$ outperforms $\texttt{BVAE}_1$ on FashionMNIST vs. MNIST and SVHN vs. CIFAR10, where in-distribution and OoD data is very distinct, but not on FashionMNIST (held-out), where the datasets are much more similar. This suggests that capturing a distribution over encoders ϕ is particularly beneficial when train and test data live on different manifolds (as overfitting ϕ is more critical), while the fixed encoder ϕ^* generalizes better when train and test manifolds are similar, which is as expected intuitively. Finally, Fig. 2 shows histograms of the log-likelihoods (middle column) and of the $\texttt{BVAE}_2$ scores (right column) on (top) FashionMNIST in-distribution (blue) vs. MNIST OoD (orange) as well as (bottom) SVHN in-distribution (blue) vs. CIFAR10 OoD (orange) test data. While the log-likelihoods strongly overlap, our proposed score more clearly separates in-distribution data (closer to the r.h.s.) from OoD data (closer to the l.h.s.).

5.2. Out-of-Distribution Detection in Latent Space

While input space OoD detection is well-studied, *latent space* OoD detection has only recently been identified as a critical open problem (Griffiths & Hernández-Lobato, 2017; Gómez-Bombarelli et al., 2018; Mahmood & Hernández-Lobato, 2019; Alperstein et al., 2019) (see also Section 1). Thus, there is a lack of suitable experimental benchmarks, making a quantitative evaluation challenging. A major issue in designing benchmarks based on commonly-used datasets such as MNIST is that it is unclear how to obtain *ground truth labels* for which latent points are OoD and which are not, as we require OoD labels for *all possible* latent points $\mathbf{z}^* \in \mathbb{Z}$, not just for those corresponding to inputs $\mathbf{x}^*$ from the given dataset. As a *first step* towards facilitating a systematic empirical evaluation of latent space OoD detection techniques, we propose the following experimental protocol.

We use the $\texttt{BVAE}_1$ variant (see Section 5.1), as latent space detection does not require encoder robustness. We train the model on FashionMNIST (or potentially any other dataset), and then sample $N = 10,000$ latent test points $\mathbf{z}^*$ from the Gaussian $\mathcal{N}(\mathbf{0}, b \cdot \mathbb{I}_d)$ where $b \in \mathbb{R}^+$ (we use $b = 10,000$), following Mahmood & Hernández-Lobato (2019). Since there do not exist ground truth labels for which latent points $\mathbf{z}^*$ are OoD or not, we compute a classifier-based *OoD proxy score* (to be detailed below) for each of the N latent test points and then simply *define* the $N/2$ latents with the lowest scores to be in-distribution, and all others to be OoD.

Table 1. AUROC↑, AUPRC↑, and FPR80↓ scores (where higher ↑ or lower ↓ is better) of our methods (top two rows) and the baselines (bottom four rows) . For the experiment on FashionMNIST with with held-out classes, we report the mean scores over all five class splits.

	FashionMNIST vs MNIST			**SVHN vs CIFAR10**			**FashionMNIST (held-out)**		
	AUROC↑	AUPRC↑	FPR80↓	AUROC↑	AUPRC↑	FPR80↓	AUROC↑	AUPRC↑	FPR80↓
$\texttt{BVAE}_1$	0.904	0.891	0.117	0.807	0.793	0.331	**0.693**	**0.680**	**0.540**
$\texttt{BVAE}_2$	**0.921**	**0.907**	**0.082**	**0.814**	**0.799**	**0.310**	0.683	0.668	0.558
LL	0.557	0.564	0.703	0.574	0.575	0.634	0.565	0.577	0.683
LLR	0.617	0.613	0.638	0.570	0.570	0.638	0.560	0.569	0.698
TT	0.482	0.502	0.833	0.395	0.428	0.859	0.482	0.496	0.806
WAIC	0.541	0.548	0.798	0.293	0.380	0.912	0.446	0.464	0.827

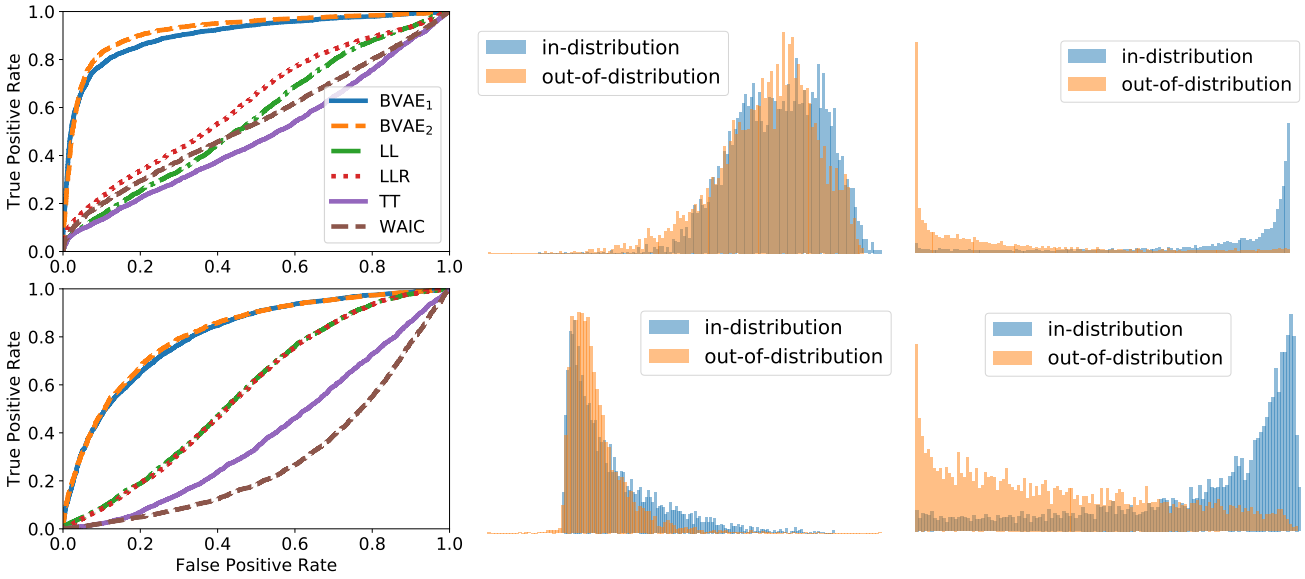

Figure 2. Results on the (top) FashionMNIST vs. MNIST and (bottom) SVHN vs. CIFAR10 benchmark. (Left) ROC curves, and histograms of the (middle) LL and (right) $\texttt{BVAE}_2$ scores. $\texttt{BVAE}_2$ separates in-distribution and OoD data much more clearly than LL.

To this end, we train an ensemble (Lakshminarayanan et al., 2017) of J convolutional NN classifiers with parameters $\mathbf{W} = \{\mathbf{w}_j\}_{j=1}^J$ on FashionMNIST. We then approximate the novelty score for discriminative models proposed by Houlsby et al. (2011), i.e., $Q_{\mathbf{W}}[\mathbf{x}^*] = \mathbb{H}\left(\frac{1}{J}\sum_{\mathbf{w}\in\mathbf{W}} p(y|\mathbf{x}^*, \mathbf{w})\right) - \frac{1}{J}\sum_{\mathbf{w}\in\mathbf{W}} \mathbb{H}(p(y|\mathbf{x}^*, \mathbf{w}))$, where the first term is the entropy of the mixture $\frac{1}{J}\sum_{\mathbf{w}\in\mathbf{W}} p(y|\mathbf{x}^*, \mathbf{w})$ of categorical distributions $p(y|\mathbf{x}^*, \mathbf{w})$ (which is again categorical with averaged probits), and the second term is the average entropy of the predictive class distribution of the classifier with parameters $\mathbf{w}$. Alternatively, one could also use the closely related OoD score $\sum_{\mathbf{w}\in\mathbf{W}} \mathrm{KL}(p(y|\mathbf{x}^*, \mathbf{w})\|p(y|\mathbf{x}^*, \mathbb{D}))$ of Lakshminarayanan et al. (2017). Since $Q_{\mathbf{W}}[\mathbf{x}^*]$ requires a test input $\mathbf{x}^*$, and we only have the latent code $\mathbf{z}^*$ corresponding to $\mathbf{x}^*$ in our setting, we instead consider the *expected* novelty under the mixture decoding distribution $p(\mathbf{x}|\mathbf{z}^*, \mathbb{D})$, $\mathbb{E}_{p(\mathbf{x}|\mathbf{z}^*,\mathbb{D})}[Q_{\mathbf{W}}[\mathbf{x}]] \simeq \frac{1}{L}\sum_{l=1}^L Q_{\mathbf{W}}[\mathbf{x}_l]$ with $\mathbf{x}_l \sim p(\mathbf{x}|\mathbf{z}^*, \mathbb{D})$. In practice, we use an ensemble of $J = 5$ classifiers and $L = 32$ input samples for the expectation.

We compare the $\texttt{BVAE}_1$ model with our *expected disagreement* score $D_\Theta[\mathbf{z}^*]$ (see Section 3.2, with $N = 32$ samples) against two baselines (which are the only existing methods we are aware of): (a) The distance of $\mathbf{z}^* \in \mathbb{R}^d$ to the spherical annulus of radius $\sqrt{d-1}$, which is where most probability mass lies under our prior $\mathcal{N}(\mathbf{0}, \mathbb{I}_d)$ (Annulus) (Alperstein et al., 2019), and (b) the log-probability of $\mathbf{z}^*$ under the *aggregated posterior* of the training data in latent space $q(\mathbf{z}) = \frac{1}{N}\sum_{\mathbf{x}\in\mathbb{D}} q(\mathbf{z}|\mathbf{x}, \phi)$, i.e., a uniform mixture of N Gaussians in our case (qz)[11] (Mahmood & Hernández-Lobato, 2019). Fig. 3 shows that our proposed method significantly outperforms the two baselines on this task.

[11]For efficiency, we only consider the 100 nearest neighbors (found by a 100-NN model) of a latent test point $\mathbf{z}^*$ for computing this log-probability (Mahmood & Hernández-Lobato, 2019).

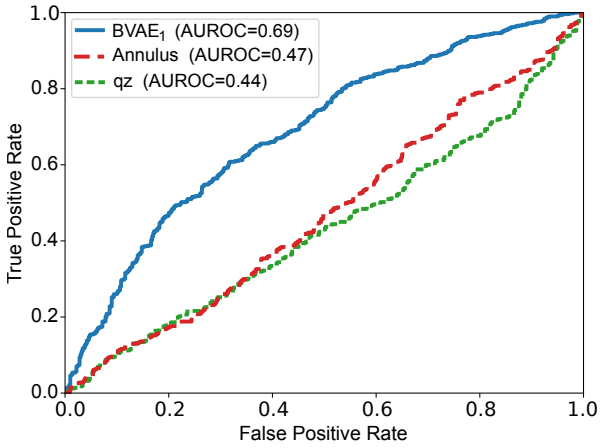

Figure 3. ROC curves on the latent space OoD benchmark, showing that our $\texttt{BVAE}_1$ significantly outperforms the other methods.

6. Related Work

Supervised/Discriminative outlier detection methods. Most existing OoD detection approaches are *task-specific* in that they are applicable within the context of a given prediction task. As described in Section 1, these approaches train a *deep discriminative model* in a *supervised* fashion using the given labels. To detect outliers w.r.t. the target task, such approaches typically rely on some sort of *confidence score* to decide on the reliability of the prediction, which is either produced by modifying the model and/or training procedure, or computed/extracted post-hoc from the model and/or predictions (An & Cho, 2015; Sölch et al., 2016; Hendrycks & Gimpel, 2016; Liang et al., 2017; Hendrycks et al., 2018; Shafaei et al., 2018; DeVries & Taylor, 2018; Sricharan & Srivastava, 2018; Ahmed & Courville, 2019). Alternatively, some methods use predictive uncertainty estimates for OoD detection (Gal & Ghahramani, 2016; Lakshminarayanan et al., 2017; Malinin & Gales, 2018; Osawa et al., 2019; Ovadia et al., 2019) (cf. Section 3.1). The main drawback of such task-specific approaches is that discriminatively trained models by design discard all input features which are not informative about the specific prediction task at hand, such that information that is relevant for general OoD detection might be lost. Thus, whenever the task changes, the predictive (and thus OoD detection) model must be re-trained from scratch, even if the input data remains the same.

Unsupervised/Generative outlier detection methods. In contrast, *task-agnostic* OoD detection methods solely use the inputs for the *unsupervised* training of a *DGM* to capture the data distribution, which makes them independent of any prediction task and thus more general. Only a few recent works fall into this category. Ren et al. (2019) propose to correct the likelihood $\log p(\mathbf{x}^*|\theta)$ for confounding general population level background statistics captured by a background model $p(\mathbf{x}^*|\theta_0)$, resulting in the score $\log p(\mathbf{x}^*|\theta) - \log p(\mathbf{x}^*|\theta_0)$. The background model $p(\mathbf{x}^*|\theta_0)$ is in practice trained by perturbing the data $\mathbb{D}$ with noise to corrupt its semantic structure, i.e., by sampling input dimensions i.i.d. from a Bernoulli distribution with rate $\mu \in [0.1, 0.2]$ and replacing their values by uniform noise, e.g. $x_i \sim \mathcal{U}\{0, \dots, 255\}$ for images. Choi & Jang (2018) propose to use an ensemble (Lakshminarayanan et al., 2017) of independently trained likelihood-based DGMs (i.e., with random parameter initializations and random data shuffling) to approximate the *Watanabe-Akaike Information Criterion* (WAIC) (Watanabe, 2010) $\mathbb{E}_{p(\theta|\mathbb{D})}[\log p(\mathbf{x}^*|\theta)] - \mathrm{Var}_{p(\theta|\mathbb{D})}[\log p(\mathbf{x}^*|\theta)]$, which provides an asymptotically correct likelihood estimate between the training and test set expectations (however, assuming a *fixed* underlying data distribution). Finally, Nalisnick et al. (2019) propose to account for the *typicality* of $\mathbf{x}^*$ via the score $\left|\log p(\mathbf{x}^*|\theta) - \frac{1}{N}\sum_{\mathbf{x}\in\mathbb{D}} \log p(\mathbf{x}|\theta)\right|$, although they focus on *batches* of test inputs instead of single inputs.

Bayesian deep generative modeling. Only a few works have tried to bring the benefits of Bayesian inference to DGMs, none of which addresses OoD detection. While Kingma & Welling (2013) describe how to do VI over the decoder parameters of a VAE (see their Appendix F), this is neither motivated nor empirically evaluated. Hernández-Lobato et al. (2016) do mean-field Gaussian VI over the encoder and decoder parameters of an importance-weighted autoencoder (Burda et al., 2015) to increase model flexibility and improve generalization performance. Nguyen et al. (2017) do mean-field Gaussian VI over the decoder parameters of a VAE to enable continual learning. Saatci & Wilson (2017) use stochastic gradient MCMC to sample the parameters of a generative adversarial network (Goodfellow et al., 2014) to increase model expressiveness. Gong et al. (2019) use stochastic gradient MCMC to sample the decoder parameters of a VAE for feature-wise active learning.

7. Conclusion

We proposed an effective method for unsupervised OoD detection, both in input space and in latent space, which uses information-theoretic metrics based on the posterior distribution over the parameters of a DGM (in particular a VAE). In the future, we want to explore extensions to other approximate inference techniques (e.g. variational inference (Blei et al., 2017)), and to other DGMs (e.g., flow-based (Kingma & Dhariwal, 2018) or auto-regressive (Van den Oord et al., 2016) DGMs). Finally, we hope that this paper will inspire many follow-up works that will (a) develop further benchmarks and methods for the underappreciated yet critical problem of *latent space* OoD detection, and (b) further explore the described paradigm of information-theoretic OoD detection, which might be a promising approach towards the grand goal of making DNNs more reliable and robust.

ACKNOWLEDGEMENTS

We would like to thank Wenbo Gong, Gregor Simm, John Bronskill, Umang Bhatt, Andrew Y. K. Foong, and the anonymous reviewers of an earlier version of this paper for their helpful suggestions. Erik Daxberger would like to thank the EPSRC, Qualcomm, and the Cambridge-Tübingen PhD Fellowship for supporting his studies.

References

Ahmed, F. and Courville, A. Detecting semantic anomalies. *arXiv preprint arXiv:1908.04388*, 2019.

Akcay, S., Atapour-Abarghouei, A., and Breckon, T. P. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Asian Conference on Computer Vision*, pp. 622–637. Springer, 2018.

Alemi, A. A., Fischer, I., and Dillon, J. V. Uncertainty in the variational information bottleneck. *arXiv preprint arXiv:1807.00906*, 2018.

Alperstein, Z., Cherkasov, A., and Rolfe, J. T. All smiles variational autoencoder. *arXiv preprint arXiv:1905.13343*, 2019.

Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., and Mané, D. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

An, J. and Cho, S. Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, 2(1), 2015.

Betancourt, M. A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434*, 2017.

Bishop, C. M. Novelty detection and neural network validation. *IEE Proceedings-Vision, Image and Signal processing*, 141(4):217–222, 1994.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

Burda, Y., Grosse, R., and Salakhutdinov, R. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*, 2015.

Chen, T., Fox, E., and Guestrin, C. Stochastic gradient hamiltonian monte carlo. In *International conference on machine learning*, pp. 1683–1691, 2014.

Choi, H. and Jang, E. Generative ensembles for robust anomaly detection. *arXiv preprint arXiv:1810.01392*, 2018.

Cremer, C., Li, X., and Duvenaud, D. Inference suboptimality in variational autoencoders. *arXiv preprint arXiv:1801.03558*, 2018.

DeVries, T. and Taylor, G. W. Learning confidence for out-of-distribution detection in neural networks. *arXiv preprint arXiv:1802.04865*, 2018.

Duane, S., Kennedy, A. D., Pendleton, B. J., and Roweth, D. Hybrid monte carlo. *Physics letters B*, 195(2):216–222, 1987.

Gal, Y. Uncertainty in deep learning. *University of Cambridge*, 1:3, 2016.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pp. 1050–1059, 2016.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

Gong, W., Tschiatschek, S., Turner, R., Nowozin, S., and Hernández-Lobato, J. M. Icebreaker: Element-wise active information acquisition with bayesian deep latent gaussian model. *arXiv preprint arXiv:1908.04537*, 2019.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. *Deep learning*, volume 1. MIT Press, 2016.

Griffiths, R.-R. and Hernández-Lobato, J. M. Constrained bayesian optimization for automatic chemical design. *arXiv preprint arXiv:1709.05501*, 2017.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1321–1330. JMLR. org, 2017.

Hendrycks, D. and Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Hendrycks, D., Mazeika, M., and Dietterich, T. G. Deep anomaly detection with outlier exposure. *arXiv preprint arXiv:1812.04606*, 2018.

Hernández-Lobato, D., Bui, T. D., Li, Y., Hernández-Lobato, J. M., and Turner, R. E. Importance weighted autoencoders with random neural network parameters. *Workshop on Bayesian Deep Learning, NIPS 2016*, 2016.

Houlsby, N., Huszár, F., Ghahramani, Z., and Lengyel, M. Bayesian active learning for classification and preference learning. *arXiv preprint arXiv:1112.5745*, 2011.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kingma, D. P. and Dhariwal, P. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems*, pp. 10215–10224, 2018.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kusner, M. J., Paige, B., and Hernández-Lobato, J. M. Grammar variational autoencoder. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 1945–1954. JMLR. org, 2017.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, pp. 6402–6413, 2017.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *nature*, 521(7553):436, 2015.

Liang, S., Li, Y., and Srikant, R. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017.

Lu, X., Gonzalez, J., Dai, Z., and Lawrence, N. Structured variationally auto-encoded optimization. In *International Conference on Machine Learning*, pp. 3273–3281, 2018.

Luo, R., Tian, F., Qin, T., Chen, E., and Liu, T.-Y. Neural architecture optimization. In *Advances in Neural Information Processing Systems*, pp. 7827–7838, 2018.

MacKay, D. J. Information-based objective functions for active data selection. *Neural computation*, 4(4):590–604, 1992.

Maddox, W. J., Izmailov, P., Garipov, T., Vetrov, D. P., and Wilson, A. G. A simple baseline for bayesian uncertainty in deep learning. In *Advances in Neural Information Processing Systems*, pp. 13132–13143, 2019.

Mahmood, O. and Hernández-Lobato, J. M. A cold approach to generating optimal samples. *arXiv preprint arXiv:1905.09885*, 2019.

Malinin, A. and Gales, M. Predictive uncertainty estimation via prior networks. In *Advances in Neural Information Processing Systems*, pp. 7047–7058, 2018.

Martino, L., Elvira, V., and Louzada, F. Effective sample size for importance sampling based on discrepancy measures. *Signal Processing*, 131:386–401, 2017.

Mattei, P.-A. and Frellsen, J. Refit your encoder when new data comes by. In *3rd NeurIPS workshop on Bayesian Deep Learning*, 2018.

Nalisnick, E., Matsukawa, A., Teh, Y. W., Gorur, D., and Lakshminarayanan, B. Do deep generative models know what they don't know? *arXiv preprint arXiv:1810.09136*, 2018.

Nalisnick, E., Matsukawa, A., Teh, Y. W., and Lakshminarayanan, B. Detecting out-of-distribution inputs to deep generative models using a test for typicality. *arXiv preprint arXiv:1906.02994*, 2019.

Nguyen, A., Yosinski, J., and Clune, J. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 427–436, 2015.

Nguyen, C. V., Li, Y., Bui, T. D., and Turner, R. E. Variational continual learning. *arXiv preprint arXiv:1710.10628*, 2017.

Osawa, K., Swaroop, S., Jain, A., Eschenhagen, R., Turner, R. E., Yokota, R., and Khan, M. E. Practical deep learning with bayesian principles. *arXiv preprint arXiv:1906.02506*, 2019.

Ovadia, Y., Fertig, E., Ren, J., Nado, Z., Sculley, D., Nowozin, S., Dillon, J. V., Lakshminarayanan, B., and Snoek, J. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *arXiv preprint arXiv:1906.02530*, 2019.

Parzen, E. On estimation of a probability density function and mode. *The annals of mathematical statistics*, 33(3): 1065–1076, 1962.

Ren, J., Liu, P. J., Fertig, E., Snoek, J., Poplin, R., DePristo, M. A., Dillon, J. V., and Lakshminarayanan, B. Likelihood ratios for out-of-distribution detection. *arXiv preprint arXiv:1906.02845*, 2019.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Saatci, Y. and Wilson, A. G. Bayesian GAN. In *Advances in neural information processing systems*, pp. 3622–3631, 2017.

Shafaei, A., Schmidt, M., and Little, J. J. Does your model know the digit 6 is not a cat? a less biased evaluation

of" outlier" detectors. *arXiv preprint arXiv:1809.04729*, 2018.

Shu, R., Bui, H. H., Zhao, S., Kochenderfer, M. J., and Ermon, S. Amortized inference regularization. In *Advances in Neural Information Processing Systems*, pp. 4393–4402, 2018.

Sölch, M., Bayer, J., Ludersdorfer, M., and van der Smagt, P. Variational inference for on-line anomaly detection in high-dimensional time series. *arXiv preprint arXiv:1602.07109*, 2016.

Springenberg, J. T., Klein, A., Falkner, S., and Hutter, F. Bayesian optimization with robust bayesian neural networks. In *Advances in Neural Information Processing Systems*, pp. 4134–4142, 2016.

Sricharan, K. and Srivastava, A. Building robust classifiers through generation of confident out of distribution examples. *arXiv preprint arXiv:1812.00239*, 2018.

Van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al. Conditional image generation with pixelcnn decoders. In *Advances in neural information processing systems*, pp. 4790–4798, 2016.

Watanabe, S. Asymptotic equivalence of bayes cross validation and widely applicable information criterion in singular learning theory. *Journal of Machine Learning Research*, 11(Dec):3571–3594, 2010.

Yin, M. and Zhou, M. Semi-implicit variational inference. *arXiv preprint arXiv:1805.11183*, 2018.

Zenati, H., Foo, C. S., Lecouat, B., Manek, G., and Chandrasekhar, V. R. Efficient gan-based anomaly detection. *arXiv preprint arXiv:1802.06222*, 2018.

A. Further Details on the Information-theoretic Perspective of our Disagreement Score

In Section 3.3, we mentioned that the disagreement score $D_\Theta[\mathbf{x}^*]$ defined in Eq. (5) can be viewed as quantifying the *informativeness* of the input $\mathbf{x}^*$ for updating the DGM parameters θ to the ones capturing the true density, yielding an information-theoretic perspective on OoD detection and revealing a close relationship to *information-theoretic active learning* (MacKay, 1992). While this connection intuitively sensible, we now further describe and justify it.

In the paradigm of *active learning*, the goal is to iteratively select inputs $\mathbf{x}^*$ which improve our estimate of the model parameters θ as rapidly as possible, in order to obtain a decent estimate of θ using as little data as possible, which is critical in scenarios where obtaining training data is expensive (e.g. in domains where humans or costly simulations have to be queried to obtain data, which includes many medical or scientific applications). The main idea of *information-theoretic* active learning is to maintain a posterior distribution $p(\theta|\mathbb{D})$ over the model parameters θ given the training data $\mathbb{D}$ observed thus far, and to then select the new input $\mathbf{x}^*$ based on its *informativeness* about the distribution $p(\theta|\mathbb{D})$, which is measured by the *change in distribution* between the current $p(\theta|\mathbb{D})$ posterior and the updated posterior $p(\theta|\mathbb{D}^*)$ with $\mathbb{D}^* = \mathbb{D} \cup \{\mathbf{x}^*\}$. This change in the posterior distribution can, for example, be quantified by the cross-entropy or KL divergence between $p(\theta|\mathbb{D})$ and $p(\theta|\mathbb{D}^*)$ (MacKay, 1992), or by the decrease in entropy between $p(\theta|\mathbb{D})$ and $p(\theta|\mathbb{D}^*)$ (Houlsby et al., 2011).

Intriguingly, while the problems of active learning and out-of-distribution detection have clearly distinct goals, they are fundamentally related in that they both critically rely on a reliable way to quantify how *different* an input $\mathbf{x}^*$ is from the training data $\mathbb{D}$ (or, put differently, how *novel* or *informative* $\mathbf{x}^*$ is). While in active learning, we aim to identify the input $\mathbf{x}^*$ that is maximally *different* (or *novel* / *informative*) in order to best improve our estimate of the model parameters by adding $\mathbf{x}^*$ to the training dataset $\mathbb{D}$, in out-of-distribution detection, we aim to classify a given input $\mathbf{x}^*$ as either in-distribution or OoD based on how *different* (or *novel* / *informative*) it is. This naturally suggests the possibility of leveraging methods to quantify the *novelty* / *informativeness* of an input $\mathbf{x}^*$ developed for one problem, and apply it to the other problem. However, most measures used in active learning are designed for *continuous* representations of the distributions $p(\theta|\mathbb{D})$ and $p(\theta|\mathbb{D}^*)$, and are not directly applicable in our setting where $p(\theta|\mathbb{D})$ and $p(\theta|\mathbb{D})$ are represented by a *discrete* set of samples Θ.

That being said, $D_\Theta[\mathbf{x}^*]$ can indeed be viewed as quantifying the *change in distribution* between the sample-based representations of $p(\theta|\mathbb{D})$ and $p(\theta|\mathbb{D}^*)$ induced by $\mathbf{x}^*$ (and thus the *informativeness* of $\mathbf{x}^*$), revealing a link to information-theoretic active learning. In particular, (Martino et al., 2017) show that $D_\Theta[\mathbf{x}^*]$ (which corresponds to the *effective sample size*, as described in Section 3.3) is closely related to the *Euclidean distance* between the vector of importance weights $\mathbf{w} = [w_\theta]_{\theta\in\Theta}$ and the vector $\mathbf{w}^* = [\frac{1}{M}]_{\theta\in\Theta}$ of probabilities defining the *discrete uniform probability mass function*, i.e.,

$$\|\mathbf{w} - \mathbf{w}^*\|_2 = \sqrt{\frac{1}{D_\Theta[\mathbf{x}^*]} - \frac{1}{M}} \quad \Longleftrightarrow \quad D_\Theta[\mathbf{x}^*] = \frac{1}{\|\mathbf{w} - \mathbf{w}^*\|_2^2 + \frac{1}{M}} \tag{8}$$

such that *maximizing* the score $D_\Theta[\mathbf{x}^*]$ is equivalent to *minimizing* the Euclidian distance $\|\mathbf{w} - \mathbf{w}^*\|_2$. Now, since

$$p(\theta|\mathbb{D}^*) = \frac{p(\mathbf{x}^*|\theta)p(\theta|\mathbb{D})}{\int p(\mathbf{x}^*|\theta)p(\theta|\mathbb{D})d\theta} = \frac{p(\mathbf{x}^*|\theta)}{p(\mathbf{x}^*|\mathbb{D})}p(\theta|\mathbb{D}) \overset{(6)}{\simeq} Mw_\theta p(\theta|\mathbb{D}) \,, \tag{9}$$

we observe that for a given model $\theta \in \Theta$, the posterior $p(\theta|\mathbb{D}^*)$ is equal to $p(\theta|\mathbb{D})$ if and only if $Mw_\theta = 1 \Longleftrightarrow w_\theta = \frac{1}{M}$, such that $p(\theta|\mathbb{D}^*)$ is equal to $p(\theta|\mathbb{D})$ for *all* models $\theta \in \Theta$ if and only if the weight vector $\mathbf{w} = [w_\theta]_{\theta\in\Theta}$ is equal to the vector $\mathbf{w}^* = [\frac{1}{M}]_{\theta\in\Theta}$ defining the discrete uniform probability mass function (pmf), in which case their Euclidian distance is minimized at $\|\mathbf{w} - \mathbf{w}^*\|_2 = 0$. As a result, the new posterior $p(\theta|\mathbb{D}^*)$ is identical to the previous posterior $p(\theta|\mathbb{D})$ over the models $\theta \in \Theta$ (i.e., the change in the posterior is *minimized*) if and only if the score $D_\Theta[\mathbf{x}^*]$ is *maximized* to be $D_\Theta[\mathbf{x}^*] = M$. Conversely, the Euclidean distance is *maximized* at $\|\mathbf{w} - \mathbf{w}^*\|_2 = \sqrt{(1 - \frac{1}{M})}$ if and only if the weight vector is $\mathbf{w} = [0, \ldots, 0, 1, 0, \ldots, 0]$, in which case the new posterior is $p(\theta|\mathbb{D}^*) = 0$ for all $M - 1$ models θ for which $w_\theta = 0$, and $p(\theta|\mathbb{D}^*) = Mp(\theta|\mathbb{D})$ for the single model θ for which $w_\theta = 1$. Thus, the change between the new and previous posterior over the models $\theta \in \Theta$ is *maximized* if and only if the score $D_\Theta[\mathbf{x}^*]$ is *minimized* to be $D_\Theta[\mathbf{x}^*] = 1$.

Finally, we observe that the notion of *change in distribution* for sample-based representations of posteriors captured by the Euclidian distance $\|\mathbf{w} - \mathbf{w}^*\|_2$ described above is *closely related* to the notion of *change in distribution* for continuous posterior representations. To see this, consider the *Kullback-Leibler (KL) divergence*, which is an information-theoretic measure for the discrepancy between distributions commonly used in information-theoretic active learning, defined as

$$\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] = \int p(\theta|\mathbb{D}) \log \frac{p(\theta|\mathbb{D})}{p(\theta|\mathbb{D}^*)} d\theta \,. \tag{10}$$

We now show that *maximizing* our proposed OoD detection score $D_\Theta[\mathbf{x}^*]$ is equivalent to *minimizing* the KL divergence $\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)]$ between the previous posterior $p(\theta|\mathbb{D})$ and the new posterior $p(\theta|\mathbb{D}^*)$, and vice versa, which is formalized in Proposition 1 below. This provides further evidence for the close connection between our proposed OoD detection approach and information-theoretic principles, and suggests that information-theoretic measures such as the KL divergence can be also used for OoD detection, yielding the paradigm of *information-theoretic out-of-distribution detection*.

Proposition 1. *Assume that the weights w_θ have some minimal, arbitrarily small, positive value $\varepsilon > 0$, i.e., $w_\theta > \varepsilon, \forall\theta \in \Theta$. Also, assume that the KL divergence $KL[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)]$ is approximated based on a set $\Theta = \{\theta_m\}_{m=1}^M$ of samples $\theta_m \sim p(\theta|\mathbb{D})$. Then, an input $\mathbf{x}^* \in \mathbb{X}$ is a* maximizer *of $D_\Theta[\mathbf{x}^*]$ if and only if it is a* minimizer *of $KL[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)]$. Furthermore, an input $\mathbf{x}^* \in \mathbb{X}$ is a* minimizer *of $D_\Theta[\mathbf{x}^*]$ if and only if it is a* maximizer *of $KL[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)]$. Formally,*

$$\arg\max_{\mathbf{x}^*\in\mathbb{X}} D_\Theta[\mathbf{x}^*] = \arg\min_{\mathbf{x}^*\in\mathbb{X}} KL[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] \,, \tag{11}$$

$$\arg\min_{\mathbf{x}^*\in\mathbb{X}} D_\Theta[\mathbf{x}^*] = \arg\max_{\mathbf{x}^*\in\mathbb{X}} KL[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] \,. \tag{12}$$

Proof. Reformulating the KL divergence in Eq. (10) and approximating it via our set Θ of posterior samples, we obtain

$$\begin{aligned}
\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] &\overset{(10)}{=} \int p(\theta|\mathbb{D}) \log \frac{p(\theta|\mathbb{D})}{p(\theta|\mathbb{D}^*)} d\theta \\
&= -\int p(\theta|\mathbb{D}) \log \frac{p(\theta|\mathbb{D}^*)}{p(\theta|\mathbb{D})} d\theta \\
&\overset{(9)}{=} -\int p(\theta|\mathbb{D}) \log \frac{p(\mathbf{x}^*|\theta)}{p(\mathbf{x}^*|\mathbb{D})} d\theta \\
&\overset{(4)}{=} -\int p(\theta|\mathbb{D}) \log \frac{p(\mathbf{x}^*|\theta)}{\mathbb{E}_{p(\theta|\mathbb{D})}[p(\mathbf{x}^*|\theta)]} d\theta \\
&= -\mathbb{E}_{p(\theta|\mathbb{D})}\left[\log \frac{p(\mathbf{x}^*|\theta)}{\mathbb{E}_{p(\theta|\mathbb{D})}[p(\mathbf{x}^*|\theta)]}\right] \\
&\simeq -\mathbb{E}_{p(\theta|\mathbb{D})}\left[\log \frac{p(\mathbf{x}^*|\theta)}{\frac{1}{M}\sum_{\theta\in\Theta} p(\mathbf{x}^*|\theta)}\right] \\
&\simeq -\frac{1}{M}\sum_{\theta\in\Theta} \log \frac{p(\mathbf{x}^*|\theta)}{\frac{1}{M}\sum_{\theta\in\Theta} p(\mathbf{x}^*|\theta)} \\
&\overset{(5)}{=} -\frac{1}{M}\sum_{\theta\in\Theta} \log M w_\theta \,.
\end{aligned} \tag{13}$$

To see that Eq. (11) holds, observe that the sample-based approximation of $\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)]$ in Eq. (13) is indeed *minimized* to be $\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] = 0$ if and only if the weight vector $\mathbf{w} = [w_\theta]_{\theta\in\Theta}$ is equal to the vector $\mathbf{w} = [\frac{1}{M}]_{\theta\in\Theta}$ defining the discrete uniform pmf (and thus if and only if the score $D_\Theta[\mathbf{x}^*]$ is *maximized* to be $D_\Theta[\mathbf{x}^*] = M$), as then

$$\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] \overset{(13)}{\simeq} -\frac{1}{M}\sum_{\theta\in\Theta} \log M w_\theta = -\frac{1}{M} M \log M \frac{1}{M} = -\log 1 = 0 \,. \tag{14}$$

We now show Eq. (12). To see why we need the assumption that the weights w_θ have some minimal, arbitrarily small, positive value $\varepsilon > 0$, i.e., $w_\theta \geq \varepsilon, \forall\theta \in \Theta$, consider the unconstrained case, where we know that $D_\Theta[\mathbf{x}^*]$ is *minimized* to be $D_\Theta[\mathbf{x}^*] = 1$ if and only if the weight vector $\mathbf{w} = [w_\theta]_{\theta\in\Theta}$ is equal to the vector $\mathbf{w} = [0, \ldots, 0, 1, 0, \ldots, 0]$. While this weight vector indeed *maximizes* the sampling-based approximation of the KL divergence to be $\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] = \infty$,

$$\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] \overset{(13)}{\simeq} -\frac{1}{M}\sum_{\theta\in\Theta} \log M w_\theta = -\frac{1}{M}[\log M + (M-1)\log 0] = \infty \,, \tag{15}$$

this maximizer is not unique, as any other weight vector containing at least one weight of $w_\theta = 0$ equally achieves the maximum of $\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] = \infty$. In theory, the KL divergence can thus not distinguish between weight vectors with different numbers of zero entries (i.e., different ℓ_0-norms $\|\mathbf{w}\|_0$), although these clearly define different degrees of change in the discrete posterior representation. However, in practice, it is very unlikely to occur that any $w_\theta = 0$. To obtain a unique

maximizer of the KL divergence, we thus assume $w_\theta \geq \varepsilon, \forall \theta \in \Theta$ (where $\varepsilon > 0$ can be chosen to be arbitrarily small), in which case $\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)]$ is maximized if and only if $\mathbf{w} = [\varepsilon, \ldots, \varepsilon, 1-(M-1)\varepsilon, \varepsilon, \ldots, \varepsilon]$, in which case

$$\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] \simeq -\frac{1}{M}\sum_{\theta\in\Theta}\log M w_\theta = -\frac{1}{M}[\log M(1-(M-1)\varepsilon) + (M-1)\log M\varepsilon] < \infty\,. \tag{16}$$

The positivity assumption thus also ensures that the KL divergence remains bounded. To see why $\mathbf{w} = [\varepsilon, \ldots, \varepsilon, 1-(M-1)\varepsilon, \varepsilon, \ldots, \varepsilon]$ maximizes the KL divergence, consider the alternative vector $\mathbf{w} = [\varepsilon, \ldots, \varepsilon, \varepsilon+\delta, \varepsilon, \ldots, \varepsilon, 1-(M-1)\varepsilon-\delta, \varepsilon, \ldots, \varepsilon]$ where any of the entries with minimal value ε is increased by some arbitrarily small, positive $\delta > 0$, such that

$$\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)] \simeq -\frac{1}{M}\sum_{\theta\in\Theta}\log M w_\theta - \frac{1}{M}[\log M(1-(M-1)\varepsilon-\delta) + (M-2)\log M\varepsilon + \log M(\varepsilon+\delta)]\,. \tag{17}$$

To see that adding such a δ decreases the value of the KL divergence, observe that Eq. (16) and Eq. (17) yield

$$\begin{aligned}
-\cancel{\tfrac{1}{M}}[\log M(1-(M-1)\varepsilon) + (M-1)\log M\varepsilon] &> -\cancel{\tfrac{1}{M}}[\log M(1-(M-1)\varepsilon-\delta) + (M-2)\log M\varepsilon + \log M(\varepsilon+\delta)] \\
\log M(1-(M-1)\varepsilon) + \cancel{(M-1)}\log M\varepsilon &< \log M(1-(M-1)\varepsilon-\delta) + \cancel{(M-2)\log M\varepsilon} + \log M(\varepsilon+\delta) \\
\log \cancel{M}(1-(M-1)\varepsilon) + \log \cancel{M}\varepsilon &< \log \cancel{M}(1-(M-1)\varepsilon-\delta) + \log \cancel{M}(\varepsilon+\delta) \\
\log(1-(M-1)\varepsilon) + \log\varepsilon &< \log(1-(M-1)\varepsilon-\delta) + \log(\varepsilon+\delta) \\
\cancel{\log}(1-(M-1)\varepsilon)\varepsilon &< \cancel{\log}(1-(M-1)\varepsilon-\delta)(\varepsilon+\delta) \\
(1-(M-1)\varepsilon)\varepsilon &< (1-(M-1)\varepsilon-\delta)(\varepsilon+\delta) \\
\varepsilon-(M-1)\varepsilon^2 &< (\varepsilon+\delta) - (M-1)\varepsilon(\varepsilon+\delta) - \delta(\varepsilon+\delta) \\
\varepsilon - M\varepsilon^2 + \varepsilon^2 &< \varepsilon + \delta - (M-1)(\varepsilon^2+\varepsilon\delta) - \varepsilon\delta - \delta^2 \\
\varepsilon - M\varepsilon^2 + \varepsilon^2 &< \varepsilon + \delta - [M\varepsilon^2 + M\varepsilon\delta - \varepsilon^2 - \varepsilon\delta] - \varepsilon\delta - \delta^2 \\
\cancel{\varepsilon} \cancel{-M\varepsilon^2} + \cancel{\varepsilon^2} &< \cancel{\varepsilon} + \delta \cancel{-M\varepsilon^2} - M\varepsilon\delta + \cancel{\varepsilon^2} + \cancel{\varepsilon\delta - \varepsilon\delta} - \delta^2 \\
0 &< \delta - M\varepsilon\delta - \delta^2 \\
0 &< 1 - M\varepsilon - \delta \\
M\varepsilon &< 1-\delta \\
\varepsilon &< \frac{1-\delta}{M}\,.
\end{aligned}$$

The condition $\varepsilon < \frac{1-\delta}{M}$ implies that the largest weight in $\mathbf{w}$, denoted by w_{θ_m}, satisfies

$$\begin{aligned}
w_{\theta_m} &= 1-(M-1)\varepsilon-\delta \\
&> 1-(M-1)\frac{1-\delta}{M} - \delta \\
&= 1-\delta-\frac{M-M\delta-1+\delta}{M} \\
&= \cancel{1-\delta-1+\delta} + \frac{1-\delta}{M} \\
&= \frac{1-\delta}{M}
\end{aligned}$$

Thus, adding an arbitrarily small δ to one of the entries of $\mathbf{w}$ indeed decreases the value of the KL divergence, except when $\varepsilon = \frac{1-\delta}{M}$, in which case the KL divergences remains the same. However, in that case, the previously largest weight becomes $w_{\theta_m} = \frac{1-\delta}{M} = \varepsilon$, yielding the weight vector $\mathbf{w} = [\frac{1-\delta}{M}, \ldots, \frac{1-\delta}{M}, \frac{1-\delta}{M}+\delta, \frac{1-\delta}{M}, \ldots, \frac{1-\delta}{M}]$, which is close to the discrete uniform pmf and thus results in a KL divergence close to zero (which is thus not relevant for characterizing the *maximizer* of the KL divergence). We can analogously identify $\mathbf{w} = [\varepsilon, \ldots, \varepsilon, 1-(M-1)\varepsilon, \varepsilon, \ldots, \varepsilon]$ to be a *minimizer* of $D_\Theta[\mathbf{x}^*]$.

To conclude, since we can choose ε to be arbitrarily small, it indeed holds that the KL divergence $\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)]$ is *minimized* if and only if the score $D_\Theta[\mathbf{x}^*]$ is *maximized*, which is when the importance weight vector defines the discrete uniform pmf, i.e., $\mathbf{w} = [\frac{1}{M}]_{\theta\in\Theta}$. Moreover, the KL divergence $\mathrm{KL}[p(\theta|\mathbb{D})\|p(\theta|\mathbb{D}^*)]$ is *maximized* if and only if the score $D_\Theta[\mathbf{x}^*]$ is *minimized*, which is when the importance weight vector is equal to $\mathbf{w} = [\varepsilon, \ldots, \varepsilon, 1-(M-1)\varepsilon, \varepsilon, \ldots, \varepsilon]$. $\square$

B. Pseudocode of BVAE Training Procedure

Pseudocode for training a *Bayesian* VAE (for both variants 1 and 2, as described in Section 4) is shown in Algorithm 2, which is contrasted to the pseudocode for training a *regular* VAE in Algorithm 1 , allowing for a direct comparison between the closely related training procedures. In particular, in Algorithm 2, the parts in purple correspond to parts that are *different* from VAE training and that apply to *both* BVAE variants 1 and 2. Furthermore, the parts in Algorithm 2 in blue correspond to BVAE *variant 1 only*, while the parts in red correspond to BVAE *variant 2 only*.

I.e., the training procedure of BVAE *variant 1* is described by the union of all black, purple and blue parts in Algorithm 2, where the only difference to the regular VAE training procedure in Algorithm 1 is that an SG-MCMC sampler is used instead of an SGD optimizer for the decoder parameters θ. The training procedure of BVAE *variant 2* is described by the union of all black, purple and red parts in Algorithm 2, where, in contrast to the regular VAE training procedure in Algorithm 1, an SG-MCMC sampler is used instead of an SGD optimizer for *both* the decoder parameters θ and the encoder parameters ϕ.

For the BVAE training procedure in Algorithm 2, we thus have to additionally specify the *burn-in length* B, which denotes the number of samples to discard at the beginning before storing any samples, as well as the *sample distance* D, which denotes the number of samples to discard in-between two subsequently stored samples (i.e., controlling the degree of correlation between the stored samples). This results in a total of $M = (T - B)/D + 1$ samples for each sampling chain.

Regular VAE training in Algorithm 1 thus produces *point estimates* θ_T for the decoder parameters and ϕ_T for the encoder parameters, while Bayesian VAE training produces a set $\Theta = \{\theta_B, \theta_{B+D}, \theta_{B+2D}, \dots, \theta_T\}$ of *posterior samples* $\phi_t \sim p(\phi|\mathbb{D})$ of decoder parameters, as well as either a *point estimate* ϕ_T for the encoder parameters (in case of variant 1), or a set $\Phi = \{\phi_B, \phi_{B+D}, \phi_{B+2D}, \dots, \phi_T\}$ of *posterior samples* $\theta_t \sim p(\theta|\mathbb{D})$ of encoder parameters (in case of variant 2).

Note that just like the regular VAE training procedure in Algorithm 1, the Bayesian VAE training procedure in Algorithm 2 can in practice be conveniently implemented by exploiting automatic differentiation tools commonly employed by modern deep learning frameworks. Finally, as SG-MCMC methods are not much more expensive to run than stochastic optimization methods (i.e., both requiring a stochastic gradient step in every iteration, but SG-MCMC potentially requiring more iterations T to generate M diverse samples), training a Bayesian VAE is not significantly more expensive than training a regular VAE.

Algorithm 1 *Regular* VAE Training

In. Dataset $\mathbb{D}$, mini-batch size $|\mathbb{M}|$, number of epochs T, generative model $p(\mathbf{x}, \mathbf{z}, \theta)$, inference model $q(\mathbf{z}|\mathbf{x}, \phi)$

Initialize ϕ_0 and θ_0
for $t = 1, \dots, T$ **do**
  Set $\hat{\phi}_0 = \phi_{t-1}$, $\hat{\theta}_0 = \theta_{t-1}$
  for $b = 1, \dots, \frac{|\mathbb{D}|}{|\mathbb{M}|}$ **do**
    Sample minibatch $\mathbb{M} \sim \mathbb{D}$
    Update $\hat{\phi}_{b-1} \to \hat{\phi}_b$ via SGD
    Update $\hat{\theta}_{b-1} \to \hat{\theta}_b$ via SGD
  end for
  Set $\phi_t = \hat{\phi}_{\frac{|\mathbb{D}|}{|\mathbb{M}|}}$, $\theta_t = \hat{\theta}_{\frac{|\mathbb{D}|}{|\mathbb{M}|}}$
end for
Out. Decoder θ_T and encoder ϕ_T

Algorithm 2 *Bayesian* VAE Training (Variant 1 & Variant 2)

In. Dataset $\mathbb{D}$, mini-batch size $|\mathbb{M}|$, number of epochs T, generative model $p(\mathbf{x}, \mathbf{z}, \theta)$, inference model $q(\mathbf{z}|\mathbf{x}, \phi)$, burn-in length B, sample distance D
Initialize ϕ_0 and θ_0, and $\Theta = \emptyset$ and $\Phi = \emptyset$
for $t = 1, \dots, T$ **do**
  Set $\hat{\phi}_0 = \phi_{t-1}$, $\hat{\theta}_0 = \theta_{t-1}$
  for $b = 1, \dots, \frac{|\mathbb{D}|}{|\mathbb{M}|}$ **do**
    Sample minibatch $\mathbb{M} \sim \mathbb{D}$
    Update $\hat{\phi}_{b-1} \to \hat{\phi}_b$ via { SGD | SG-MCMC }
    Update $\hat{\theta}_{b-1} \to \hat{\theta}_b$ via SG-MCMC
  end for
  Set $\phi_t = \hat{\phi}_{\frac{|\mathbb{D}|}{|\mathbb{M}|}}$, $\theta_t = \hat{\theta}_{\frac{|\mathbb{D}|}{|\mathbb{M}|}}$
  if $t \geq B$ and $(t - B) \bmod D = 0$ **then**
    Add $\Theta = \Theta \cup \{\theta_t\}$ and $\Phi = \Phi \cup \{\phi_t\}$
  end if
end for
Out. Decoder samples Θ and encoder { ϕ_T | samples Φ }

C. Additional Plots for Experiments

We show additional plots for the experiments conducted in Section 5. In particular, we report precision-recall curves for all benchmarks, as well as ROC curves for the FashionMNIST (held-out classes) benchmark. We show both types of precision-recall curves, depending on whether in-distribution data are considered as the false class (denoted by "in"), or whether OoD data are considered to be the false class (denoted by "out"). Fig. 4 also shows examples from the FashionMNIST dataset, in order to help visualize the different class splits for the FashionMNIST (held-out class) benchmark.

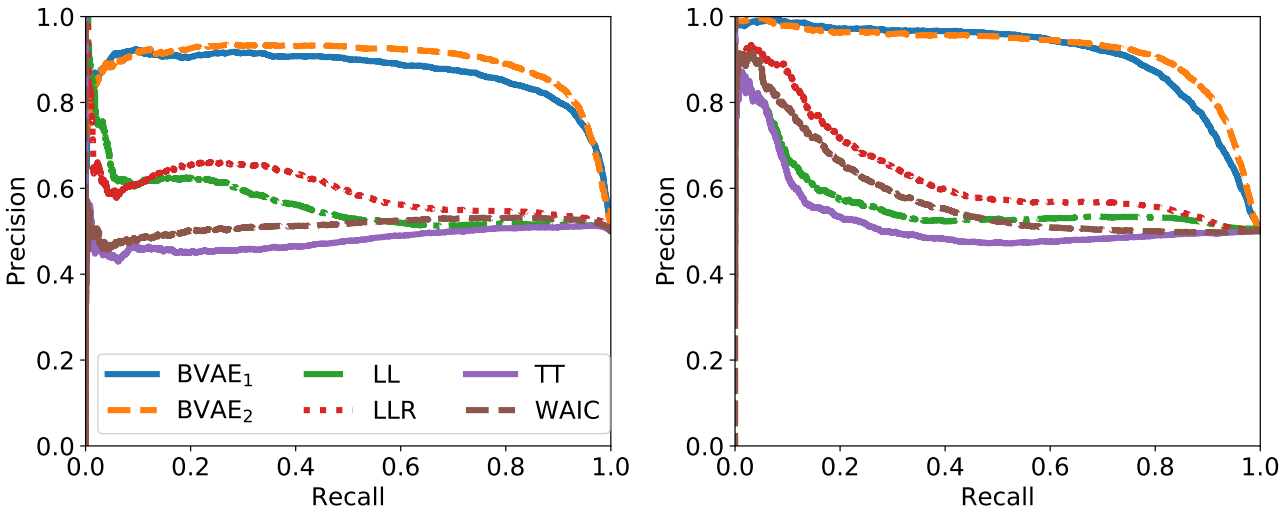

Figure 4. Precision-recall curves (in) and (out) on the FashionMNIST vs. MNIST benchmark.

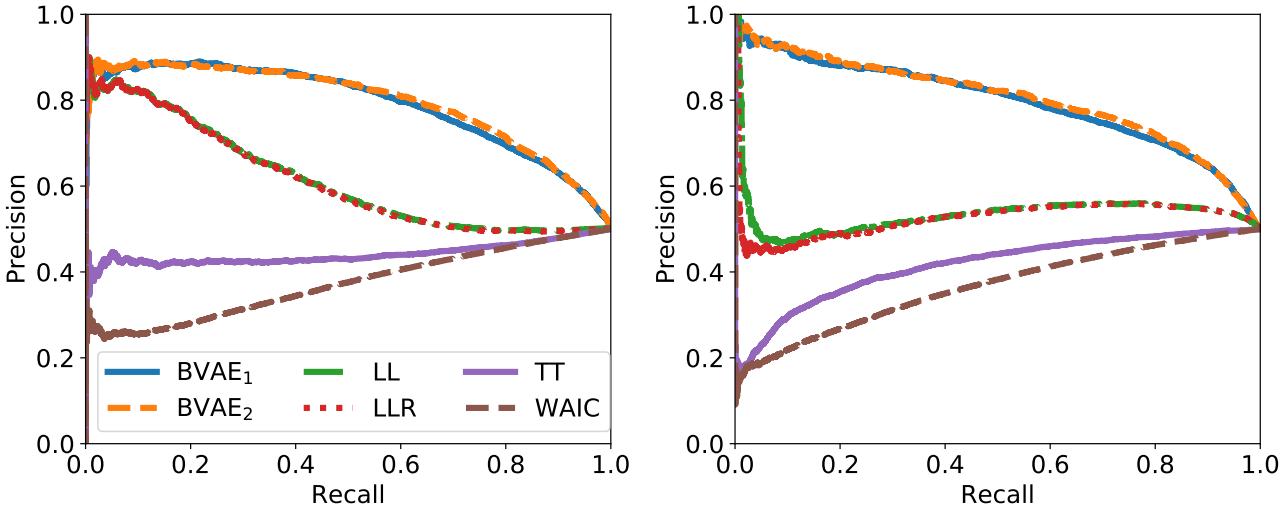

Figure 5. Precision-recall curves (in) and (out) on the SVHN vs. CIFAR10 benchmark.

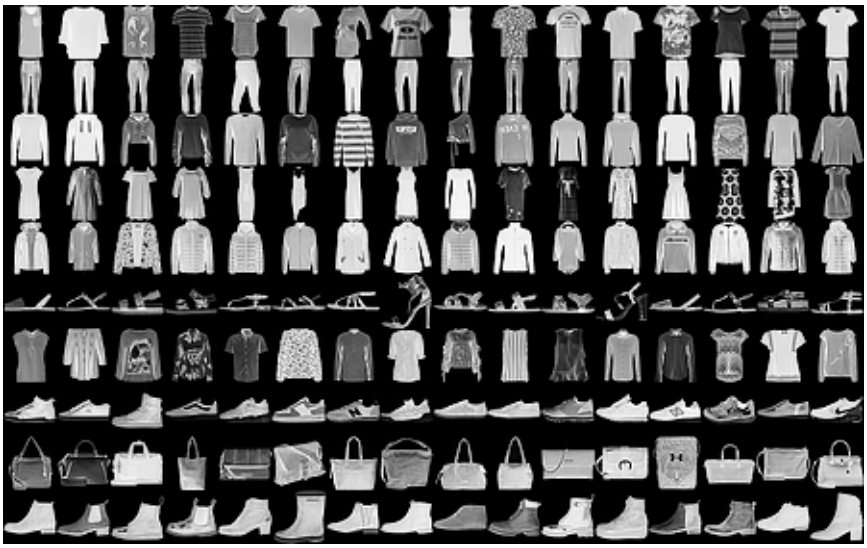

Figure 6. Examples from the FashionMNIST dataset for classes (from top to bottom) zero (t-shirt/top), one (trouser), two (pullover), three (dress), four (coat), five (sandal), six (shirt), seven (sneaker), eight (bag), and nine (ankle boot).

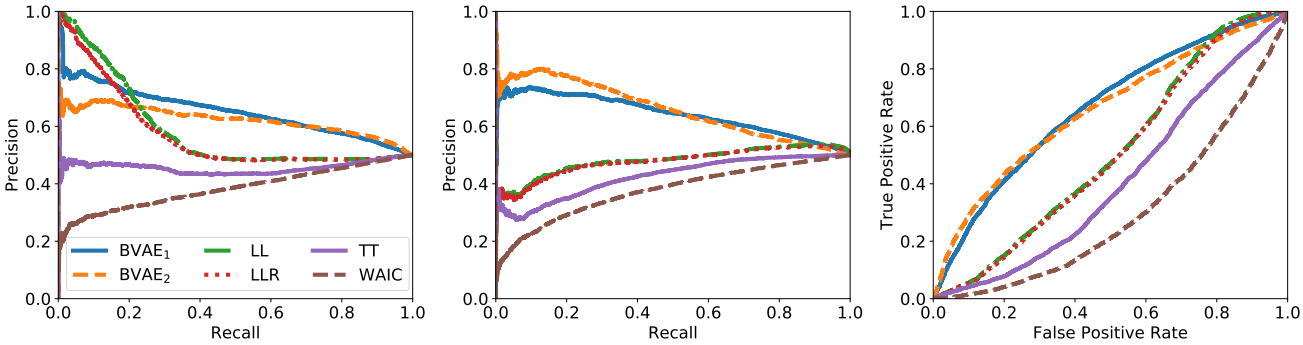

Figure 7. (Left) Precision-recall curves and (right) ROC curves of all methods on the FashionMNIST (held-out classes) benchmark with classes zero (t-shirt/top) and one (trouser) held-out.

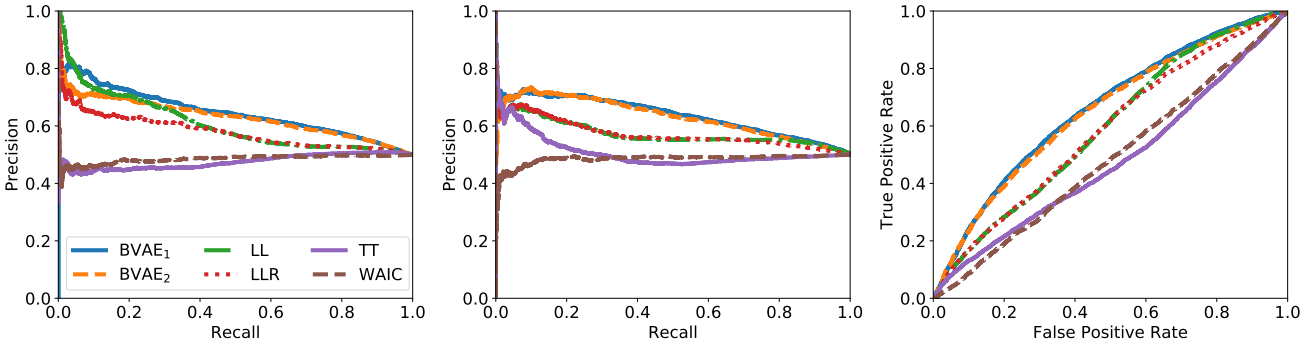

Figure 8. (From left to right) Precision-recall curves (in and out) and ROC curves of all methods on the FashionMNIST (held-out classes) benchmark with classes two (pullover) and three (dress) held-out.

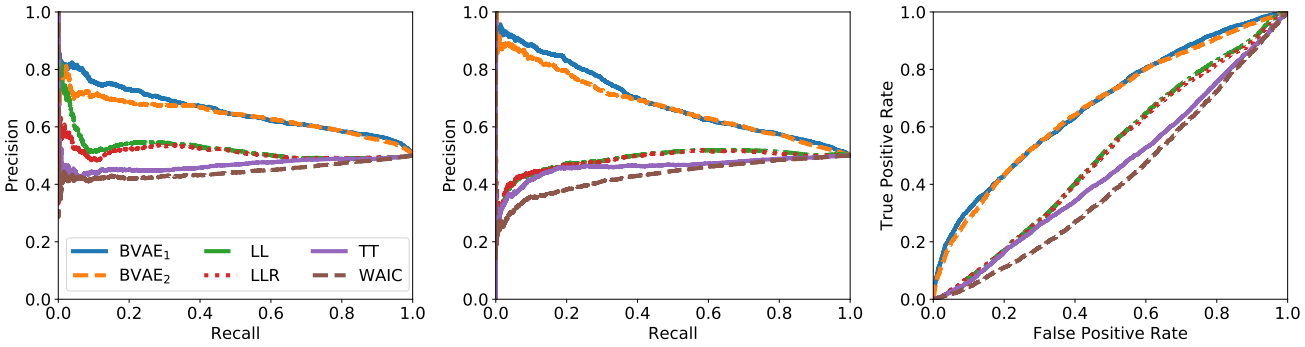

Figure 9. (From left to right) Precision-recall curves and ROC curves of all methods on the FashionMNIST (held-out classes) benchmark with classes four (coat) and five (sandal) held-out.

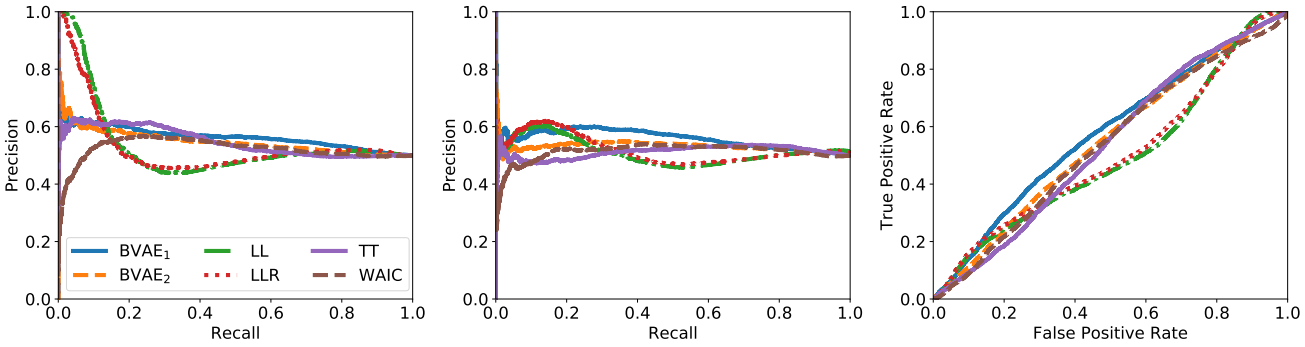

Figure 10. (From left to right) Precision-recall curves (in and out) and ROC curves of all methods on the FashionMNIST (held-out classes) benchmark with classes six (shirt) and seven (sneaker) held-out.

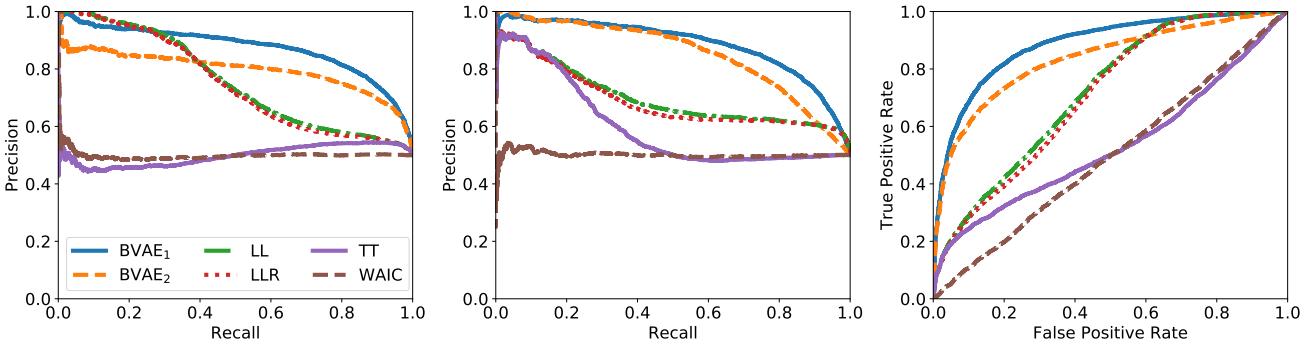

Figure 11. (From left to right) Precision-recall curves (in and out) and ROC curves of all methods on the FashionMNIST (held-out classes) benchmark with classes eight (bag) and nine (ankle boot) held-out.