

Bayesian Autoencoders: Analysing and Fixing the Bernoulli likelihood for Out-of-Distribution Detection

Bang Xiang Yong¹ Tim Pearce¹ Alexandra Brintrup¹

Abstract

After an autoencoder (AE) has learnt to reconstruct one dataset, it might be expected that the likelihood on an out-of-distribution (OOD) input would be low. This has been studied as an approach to detect OOD inputs. Recent work showed this intuitive approach can fail for the dataset pairs FashionMNIST vs MNIST. This paper suggests this is due to the use of Bernoulli likelihood and analyses why this is the case, proposing two fixes: 1) Compute the *uncertainty* of likelihood estimate by using a Bayesian version of the AE. 2) Use alternative distributions to model the likelihood.

1. Introduction

Recent works by (Choi et al., 2018; Daxberger & Hernández-Lobato, 2019; Nalisnick et al., 2019; Ren et al., 2019) reported the unreliability of using the likelihood of generative models for out-of-distribution (OOD) detection. As such, many workarounds were developed such as typicality (Nalisnick et al., 2019), likelihood ratio (Ren et al., 2019) and Watanabe-Akaike Information Criterion (WAIC) (Choi et al., 2018).

This unreliability is surprising, since autoencoders (AEs) have been extensively studied for OOD detection (also termed ‘anomaly detection’) (Basora et al., 2019; Dong & Japkowicz, 2018; Kiran et al., 2018). Intuitively, the idea is to train a neural network to reconstruct a set of training inputs, and during test time, we expect a higher ‘reconstruction loss’ on inputs which deviate from the data distribution it was trained on. The likelihood is a probabilistic measure of the reconstruction loss, representing how well the input is recovered.

Figure 1 shows a VAE trained on FashionMNIST and tested

¹Institute for Manufacturing, University of Cambridge, United Kingdom. Correspondence to: Bang Xiang Yong <bxy20@cam.ac.uk>.

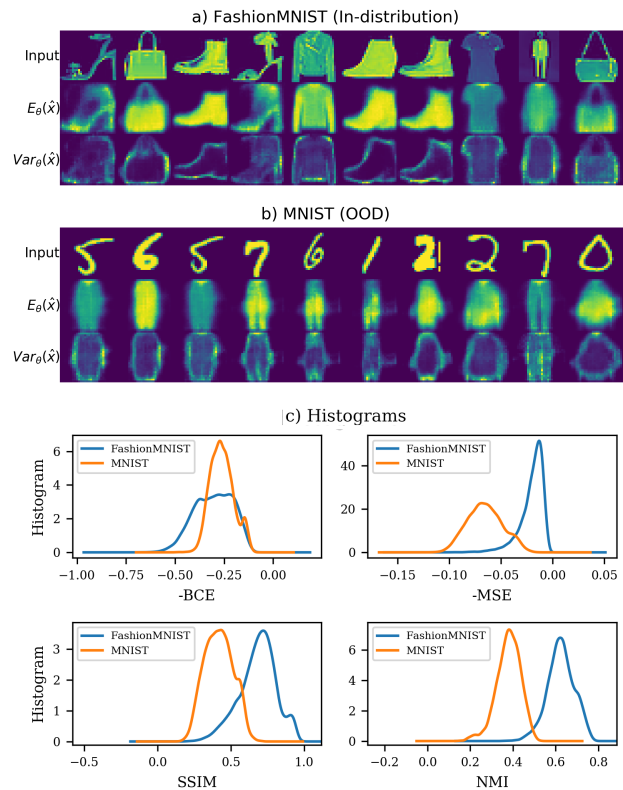


Figure 1. Mean of reconstructed signal $\mathbb{E}_{\theta}(\hat{x})$, and epistemic uncertainty of reconstructed signal $Var_{\theta}(\hat{x}^*)$, on a) FashionMNIST (in-distribution) and b) MNIST (OOD). The reconstructed images of OOD data are qualitatively different from the input images as the model only knows about fashion images. Panel c) depicts histograms of image similarity measures: negative binary cross entropy (-BCE), negative mean-squared error (-MSE), structural similarity index (SSIM) and normalised mutual information (NMI). The higher the value, the more similar the reconstructed image is to the reference image. The samples are outputs from a VAE trained under Bernoulli likelihood.

on both FashionMNIST and MNIST, and we observe the quality of reconstructed images of in-distribution and OOD data appears to be different; the model trained on Fashion-MNIST reconstructs what it knows (fashion images) when fed with digit images, and yet, the binary cross entropy (BCE)¹ is unable to tell the difference and assigns overlapping similarity score in Figure 1c). In contradiction, other well-studied image similarity measures (Wang et al., 2004) assign lower scores on the OOD images. This implies the BCE is an unreliable measure for image similarity.

Motivated by this observation, we suggest that the culprit is in the likelihood. Specifically, we find recent papers which reported poor experimental results on OOD detection have used the Bernoulli likelihood in their variational autoencoder (VAE). Although using the Bernoulli likelihood is not coherent within a probabilistic framework for image data (support of likelihood is $\{0,1\}$ instead of $[0,1]$ for each pixel value), this practice is common. For this reason, (Loaiza-Ganem & Cunningham, 2019) developed a fix, by including a normalisation constant and called this the Continuous Bernoulli distribution. Another widely used option is the Gaussian likelihood. Despite its support being not coherent with image data, empirically we find it to perform well for OOD.

In this paper, we develop Bayesian Autoencoders (BAE) and evaluate them on OOD detection using image dataset pairs which were shown to fail in recent studies. The contributions of this paper are:

1. We show the poor OOD detection performance of Bernoulli likelihood is due to confounding by proportion of zeros in an image. Likewise, the Continuous Bernoulli likelihood suffers from similar issue. Furthermore, we demonstrate that BCE is an unreliable image similarity measure compared to other well-known image similarity measures.
2. We motivate the development of the BAE for OOD detection, through various techniques to approximately sample from the posterior over weights. Although these methods have been explored with supervised BNNs, they are relatively understudied with AEs which are unsupervised models.
3. We propose two simple ways fixes for the poor performance of Bernoulli likelihood for OOD detection on image datasets; 1) for stochastic AEs such as VAEs and BAEs, compute the uncertainty of log-likelihood estimate as the OOD score, and 2) use alternative distribution to model the likelihood.

¹The Bernoulli log-likelihood is equivalent to the binary cross entropy function.

2. Methods

The AE is a neural network which maps a given set of unlabelled training data; $X = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_N\}$, $\mathbf{x}_i \in \mathbb{R}^D$ into a set of reconstructed signals, $\hat{X} = \{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \hat{\mathbf{x}}_3, \dots, \hat{\mathbf{x}}_N\}$, $\hat{\mathbf{x}}_i \in \mathbb{R}^D$. An AE is parameterised by θ , and consists of two parts: an encoder f_{encoder} , for mapping input data \mathbf{x} to a latent embedding, and a decoder f_{decoder} for mapping the latent embedding to a reconstructed signal of the input $\hat{\mathbf{x}}$ (i.e. $\hat{\mathbf{x}} = f_{\theta}(\mathbf{x}) = f_{\text{decoder}}(f_{\text{encoder}}(\mathbf{x}))$) (Goodfellow et al., 2016).

Bayes' rule can be applied to the parameters of the AE, to create a BAE,

$$p(\theta|X) = \frac{p(X|\theta) p(\theta)}{p(X)}, \quad (1)$$

where $p(X|\theta)$ is the likelihood and $p(\theta)$ is the prior distribution of the AE parameters. During prediction with test data \mathbf{x}^* , we are interested in the posterior predictive distribution, given by:

$$p(\hat{\mathbf{x}}^*|\mathbf{x}^*, X) = \int p(\hat{\mathbf{x}}^*|\mathbf{x}^*, \theta) p(\theta|X) d\theta \quad (2)$$

Since Equations 1 and 2 are analytically intractable for a deep neural network, various approximate methods have been developed such as Stochastic Gradient Markov Chain Monte Carlo (SGHMC) (Chen et al., 2014), MC-Dropout (Gal & Ghahramani, 2016), Bayes by Backprop (Blundell et al., 2015), and anchored ensembling (Pearce et al., 2020) to sample from the posterior distribution. In contrast, a deterministic AE has its parameters estimated using maximum likelihood estimation (MLE) or maximum a posteriori (MAP) when regularisation is introduced. The VAE (Kingma & Welling, 2013) and BAE are AEs formulated differently within a probabilistic framework - in the VAE, only the latent embedding is stochastic while the f_{encoder} and f_{decoder} are deterministic while the BAE (similar to BNN) has distributions over all parameters of f_{encoder} and f_{decoder} . The log-likelihood for a Bernoulli distribution is,

$$\log p(\mathbf{x}|\theta) = \frac{1}{D} \sum_{i=1}^D x_i \cdot \log \hat{x}_i + (1 - x_i) \cdot \log (1 - \hat{x}_i), \quad (3)$$

and for a diagonal Gaussian distribution,

$$\log p(\mathbf{x}|\theta) = -\left(\frac{1}{D} \sum_{i=1}^D \frac{1}{2\sigma_i^2} (x_i - \hat{x}_i)^2 + \frac{1}{2} \log \sigma_i^2\right) \quad (4)$$

For a diagonal isotropic Gaussian likelihood with variance 1, minimising the mean-squared error (MSE) function is equivalent to minimising the negative log-likelihood (NLL).

Due to the intractability of Equation 2, we obtain a set of realised samples $\{\hat{\theta}_t\}_{t=1}^T$ from the posterior over weights

and in a similar fashion to BNNs, we estimate the predictive mean $\mathbb{E}_\theta(\hat{\mathbf{x}}^*)$ and the epistemic uncertainty of reconstructed signal $\text{Var}_\theta(\hat{\mathbf{x}}^*)$ as:

$$\mathbb{E}_\theta(\hat{\mathbf{x}}^*) = \frac{1}{T} \sum_{t=1}^T f_{\hat{\theta}_t}(\mathbf{x}^*) \quad (5)$$

$$\text{Var}_\theta(\hat{\mathbf{x}}^*) = \frac{1}{T} \sum_{t=1}^T (f_{\hat{\theta}_t}(\mathbf{x}^*) - \mathbb{E}_\theta(\hat{\mathbf{x}}^*))^2 \quad (6)$$

Furthermore, we can compute the mean and variance of the log-likelihood estimate, $\mathbb{E}_\theta(\log p(\mathbf{x}^*|\theta))$ and $\text{Var}_\theta(\log p(\mathbf{x}^*|\theta))$. The WAIC (Choi et al., 2018) is given by

$$\text{WAIC}(\mathbf{x}^*) = \mathbb{E}_\theta(\log p(\mathbf{x}^*|\theta)) - \text{Var}_\theta(\log p(\mathbf{x}^*|\theta)) \quad (7)$$

3. Experimental Setup

3.1. Models

In our experiments, we train variants of AEs; deterministic AE, VAE and BAE². For the BAE, we test several inference methods - MC-Dropout, Bayes by Backprop and anchored ensembling, always assuming a diagonal isotropic Gaussian prior over weights. For all AEs, we trial three likelihoods - Bernoulli, Continuous Bernoulli or diagonal Gaussian with fixed variance 1. To tune the learning rate, we run a learning rate finder and employ a cyclic learning rate (Smith, 2017) during training of 20 epochs. For more details on the implementation, refer to Appendix A.

3.2. Methods for OOD Detection

We consider several methods to detect OOD inputs. For each AE type and likelihood, we compute $\text{Var}_\theta(\hat{\mathbf{x}}^*)$, expected value of log-likelihood $\mathbb{E}_\theta(\text{LL})$ ³, epistemic uncertainty of log-likelihood estimation, $\text{Var}_\theta(\text{LL})$ and WAIC on both the in-distribution and OOD test data.

4. Results and Discussion

In this section, firstly, following the evaluation by (Ren et al., 2019), we analyse the confoundedness of each method due to the proportion of zeros in an image. Then, we evaluate and compare the methods of detecting OOD inputs using the area under the receiver operating curve (AUROC), a threshold agnostic measure which is common in recent literature (Choi et al., 2018; Daxberger & Hernández-Lobato, 2019; Nalisnick et al., 2019; Ren et al., 2019).

Confoundedness. We now discuss an issue with using the Bernoulli likelihood as a method for OOD detection. Figure 2 plots the Bernoulli (Equation 3), Continuous Bernoulli

(Loaiza-Ganem & Cunningham, 2019) and Gaussian (Equation 4) log-likelihood for varying x , when $x = \hat{x}$ (i.e. the maximum value for a perfect reconstruction of the input). For the Bernoulli and Continuous Bernoulli likelihoods, we observe that the maximum value available is influenced by the value of x , with the highest log-likelihood available for when $x = 0$ or $x = 1$.

This is a crucial observation. MNIST contains images with a high proportion of zeros, which means high log-likelihood scores are available. Conversely, FashionMNIST images have low proportion of zeros and therefore, low log-likelihood scores are available. This property is not exhibited by Gaussian log-likelihood, which maximum value remains constant when x is varied.

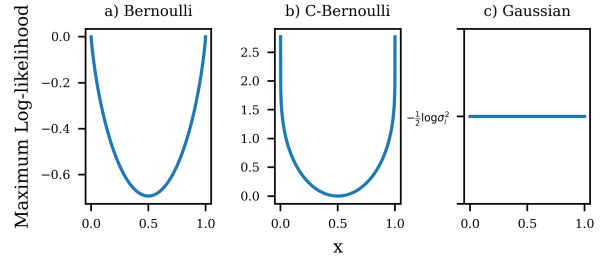


Figure 2. Maximum of a) Bernoulli and b) Continuous Bernoulli log-likelihood is influenced by the value of input, x . Conversely, c) maximum of Gaussian log-likelihood stays constant with respect to x .

Furthermore, by plotting the correlation between proportion of zeros against $\mathbb{E}_\theta(\text{LL})$ and $\text{Var}_\theta(\text{LL})$ in Figure 3, we find both the Bernoulli log-likelihood and its continuous variant are confounded by the proportion of zeros in the image (Pearson Correlation Coefficient 0.81, 0.87 respectively), in which the MNIST images have a larger proportion of zeros due to its background. This means the model assigns higher log-likelihoods to images with higher proportion of zeros, regardless of whether it is OOD input or not. In contrast, the $\mathbb{E}_\theta(\text{LL})$ for Gaussian likelihood and the $\text{Var}_\theta(\text{LL})$ do not suffer from such confoundedness.

Poor OOD detection using Bernoulli likelihood. Based on the results in Table 1, by using the $\mathbb{E}_\theta(\text{LL})$ method, the performance is poorer for all models with Bernoulli likelihood ($\text{AUROC} < 0.5$), compared to that of Gaussian likelihood ($\text{AUROC} > 0.9$)⁴. We find the Continuous Bernoulli likelihood is unable to fix this poor performance.

Combining epistemic uncertainty and likelihood im-

²Code available: github.com/bangxiangyong/bae-ood-images

³For brevity, we denote $\text{LL} = \log p(\mathbf{x}^*|\theta)$

⁴Asymmetrically, however, the Bernoulli likelihood does perform well on MNIST vs FashionMNIST (Appendix C, Table 3) which aligns with (Nalisnick et al., 2019)

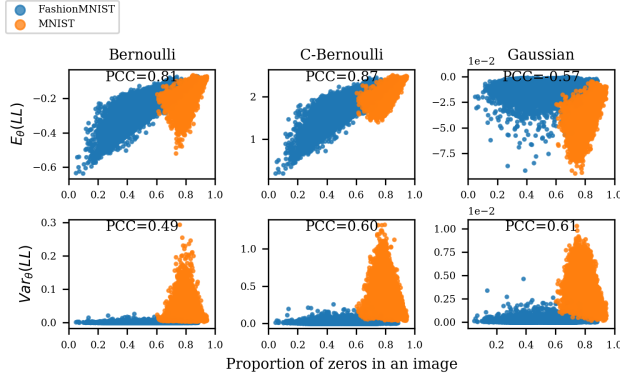


Figure 3. Relationship between $\mathbb{E}_\theta(\text{LL})$ (first row), $\text{Var}_\theta(\text{LL})$ (second row) and proportion of zeros in an image. Pearson correlation coefficient (PCC) for each plot is also shown. (FashionMNIST vs MNIST with BAE, Ensemble)

Table 1. (FashionMNIST vs MNIST) AUROC for detecting OOD inputs with deterministic AE, VAE and BAE (MC-Dropout, Bayes by Backprop and anchored ensembling) with posterior approximated under various techniques. For each likelihood- Bernoulli, Continuous Bernoulli and Gaussian, we compare the performance of using $\text{Var}_\theta(\hat{\mathbf{x}})$, $\mathbb{E}_\theta(\text{LL})$, $\text{Var}_\theta(\text{LL})$, and WAIC. Results where AUROC > 0.8 are in bold.

Model	Likelihood	AUROC			
		$\text{Var}_\theta(\hat{\mathbf{x}}^*)$	$\mathbb{E}_\theta(\text{LL})$	$\text{Var}_\theta(\text{LL})$	WAIC
Deterministic AE	Ber($\hat{\mathbf{x}}^*$)	-	0.230	-	-
Deterministic AE	C-Ber($\hat{\mathbf{x}}^*$)	-	0.203	-	-
Deterministic AE	N($\hat{\mathbf{x}}^*, 1$)	-	0.978	-	-
VAE	Ber($\hat{\mathbf{x}}^*$)	0.629	0.393	0.982	0.454
VAE	C-Ber($\hat{\mathbf{x}}^*$)	0.365	0.301	0.995	0.389
VAE	N($\hat{\mathbf{x}}^*, 1$)	0.420	0.987	0.948	0.987
BAE, MC-Dropout	Ber($\hat{\mathbf{x}}^*$)	0.610	0.224	0.983	0.250
BAE, MC-Dropout	C-Ber($\hat{\mathbf{x}}^*$)	0.670	0.183	0.981	0.237
BAE, MC-Dropout	N($\hat{\mathbf{x}}^*, 1$)	0.820	0.977	0.991	0.978
BAE, BayesBB	Ber($\hat{\mathbf{x}}^*$)	0.736	0.258	0.987	0.371
BAE, BayesBB	C-Ber($\hat{\mathbf{x}}^*$)	0.750	0.204	0.991	0.308
BAE, BayesBB	N($\hat{\mathbf{x}}^*, 1$)	0.701	0.945	0.941	0.951
BAE, Ensemble	Ber($\hat{\mathbf{x}}^*$)	0.990	0.230	0.999	0.314
BAE, Ensemble	C-Ber($\hat{\mathbf{x}}^*$)	0.978	0.191	0.999	0.418
BAE, Ensemble	N($\hat{\mathbf{x}}^*, 1$)	0.995	0.976	0.998	0.980

proves performance. On its own, $\text{Var}_\theta(\hat{\mathbf{x}}^*)$ does not perform reliably for OOD detection, with the exception of anchored ensembling. This may be due to the quality of uncertainty which was reported to be poorer with MC-Dropout and variational inference, in comparison to ensembling (Yao et al., 2019; Pearce et al., 2020). In spite of the report by (Nalisnick et al., 2019) that ensembling is not robust towards OOD, we posit this would not happen if a Gaussian likelihood or the uncertainty estimate (either $\text{Var}_\theta(\hat{\mathbf{x}}^*)$ or $\text{Var}_\theta(\text{LL})$) was used.

Surprisingly, when we combine the epistemic uncertainty and likelihood, which we call as the uncertainty of log-

likelihood estimate, $\text{Var}_\theta(\text{LL})$, this method obtains good performance (AUROC > 0.9) for all models (including VAE, BAE-MCDropout, and BAE-BayesBB which epistemic uncertainty performed poorly on their own). Moreover, for models which used Bernoulli likelihood, $\text{Var}_\theta(\text{LL})$ performed far better than $\mathbb{E}_\theta(\text{LL})$. This prevails even with the Continuous Bernoulli likelihood.

Since $\text{Var}_\theta(\text{LL})$ performs well on its own, we question the need for WAIC since the AUROC using WAIC lies between that of $\mathbb{E}_\theta(\text{LL})$ and $\text{Var}_\theta(\text{LL})$ in most of the results. Moreover, different from the implementation of (Choi et al., 2018) with the VAE, we obtain the epistemic uncertainty through importance sampling of a single model instead of an ensemble, which we find to be sufficient.

For SVHN vs CIFAR10, we obtain good AUROC with one or more models (Appendix B, Table 4), however, we are unable to obtain the same for CIFAR10 vs SVHN and we offer a possible explanation in Appendix C.

5. Related Work

(Choi et al., 2018; Daxberger & Hernández-Lobato, 2019; Nalisnick et al., 2019; Ren et al., 2019) reported the poor performance of using the log-likelihood method for OOD detection on FashionMNIST vs MNIST, which we find is due to choosing the Bernoulli distribution as likelihood, as the Gaussian likelihood do perform well on the same task. Although (Ren et al., 2019) first addressed the confoundedness of the likelihood, our work extends it by showing it is specific to the Bernoulli likelihood and conversely, the Gaussian likelihood and $\text{Var}_\theta(\text{LL})$ do not suffer from such issue. Our work is complementary to (Daxberger & Hernández-Lobato, 2019) who developed the Bayesian Variational Autoencoder (BVAE) with SGHMC, which we extend by exploring other approximation methods (MC-Dropout, Bayes by Backprop, anchored ensembling) common in BNN literature.

6. Conclusion

This paper investigated the reported inability of generative models to detect certain OOD inputs. We showed that this can be attributed to a problematic combination of the Bernoulli likelihood with images containing a high proportion of zeros.

The simplest fix for this is to switch to a Gaussian likelihood. Though the support of this distribution doesn’t match the domain of pixel values, it nonetheless proved surprisingly effective (AUROC for all AEs improved from around 0.30 to 0.98). The second fix is to use the uncertainty of log-likelihood (for VAE and BAE). With the best inference techniques (anchored ensembling), AUROC was increased to 0.99.

Acknowledgements

The research presented was supported by EMPIR (European Metrology Programme for Innovation and Research) under the MET4FOF (Metrology for the Factory of the Future) project, as well as the PITCH-IN (Promoting the Internet of Things via Collaborations between HEIs and Industry) project funded by Research England.

References

- Basora, L., Olive, X., and Dubot, T. Recent advances in anomaly detection methods applied to aviation. *Aerospace*, 6(11):117, 2019.
- Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. Weight uncertainty in neural network. In *International Conference on Machine Learning*, pp. 1613–1622, 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., Jang, E., and Alemi, A. A. Waic, but why? generative ensembles for robust anomaly detection. *arXiv preprint arXiv:1810.01392*, 2018.
- Daxberger, E. and Hernández-Lobato, J. M. Bayesian variational autoencoders for unsupervised out-of-distribution detection. *arXiv preprint arXiv:1912.05651*, 2019.
- Dong, Y. and Japkowicz, N. Threaded ensembles of autoencoders for stream learning. *Computational Intelligence*, 34(1):261–281, 2018.
- 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.
- Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. MIT Press, 2016.
- Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.
- Kiran, B. R., Thomas, D. M., and Parakkal, R. An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos. *Journal of Imaging*, 4(2):36, 2018.
- Loaiza-Ganem, G. and Cunningham, J. P. The continuous bernoulli: fixing a pervasive error in variational autoencoders. In *Advances in Neural Information Processing Systems*, pp. 13266–13276, 2019.
- Nalisnick, E., Matsukawa, A., Teh, Y. W., Gorur, D., and Lakshminarayanan, B. Do deep generative models know what they don’t know? In *International Conference on Learning Representations*, 2019.
- Pearce, T., Zaki, M., Brintrup, A., Anastassacos, N., and Neely, A. Uncertainty in neural networks: Bayesian ensembling. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2020.
- Ren, J., Liu, P. J., Fertig, E., Snoek, J., Poplin, R., Deprieto, M., Dillon, J., and Lakshminarayanan, B. Likelihood ratios for out-of-distribution detection. In *Advances in Neural Information Processing Systems*, pp. 14680–14691, 2019.
- Smith, L. N. Cyclical learning rates for training neural networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 464–472. IEEE, 2017.
- Wang, Z., Bovik, A. C., Sheikh, H. R., and Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.
- Yao, J., Pan, W., Ghosh, S., and Doshi-Velez, F. Quality of uncertainty quantification for bayesian neural network inference. *arXiv preprint arXiv:1906.09686*, 2019.

A. Implementation details

Our code is available at github.com/bangxiangyong/baeood-images for reproducibility. We use the default dataset splits provided by *torchvision*⁵. We test the trained model on the test split for both in-distribution and OOD inputs.

A.1. Network architectures

For each AE, we control the network architecture for a fair comparison. We use leaky ReLu as the activation function for each network in our experiments with a slope of 0.01. We apply the sigmoid function on the outputs. The latent size for FashionMNIST, MNIST, SVHN and CIFAR10 (in-distribution) datasets are 20, 20, 50 and 100 respectively.

Table 2. Encoder architectures for FashionMNIST, MNIST, CIFAR10 and SVHN acting as in-distribution training dataset. The decoder is a reflection of the encoder, in which the convolution layers are replaced by convolution transpose layers.

a) FashionMNIST and MNIST			
Layer	Output dimensions	Kernel	Strides
Convolution	13 x 13 x 32	4 x 4	2 x 2
Convolution	10 x 10 x 64	4 x 4	1 x 1
Reshape	6400	-	-
Dense	Latent size	-	-
b) CIFAR10 and SVHN			
Layer	Output dimensions	Kernel	Strides
Convolution	15 x 15 x 32	4 x 4	2 x 2
Convolution	12 x 12 x 64	4 x 4	1 x 1
Convolution	5 x 5 x 128	4 x 4	2 x 2
Reshape	3200	-	-
Dense	Latent size	-	-

A.2. Learning rate

We run a learning rate finder and employ a sawtooth cyclic learning rate protocol (Smith, 2017) with Adam optimiser. With a batch size of 100, the training lasts for 20 epochs. We swept the value for scaling the regularisation term over values of {10, 2, 1, 0.1, 0.01, 0.001} for each combination of model and likelihood.

A.3. Number of samples

We train 5 independently initialised AEs for BAE-Ensemble. With the VAE, BAE-MCDropout and BAE-BayesBB, we draw 100 samples from the posterior for each input image.

B. Additional results

Results for other dataset pairs (MNIST vs FashionMNIST, SVHN vs CIFAR10, and CIFAR10 vs SVHN) are shown here.

Table 3. (MNIST vs FashionMNIST) AUROC for various models and likelihoods. Results where AUROC > 0.8 are in bold.

Model	Likelihood	AUROC			
		$\text{Var}_\theta(\hat{\mathbf{x}}^*)$	$\mathbb{E}_\theta(\text{LL})$	$\text{Var}_\theta(\text{LL})$	WAIC
Deterministic AE	Ber(\hat{x}^*)	-	1.00	-	-
Deterministic AE	C-Ber(\hat{x}^*)	-	1.00	-	-
Deterministic AE	N($\hat{x}^*, 1$)	-	0.999	-	-
VAE	Ber(\hat{x}^*)	0.996	1.00	1.00	1.00
VAE	C-Ber(\hat{x}^*)	0.990	1.00	0.997	1.00
VAE	N($\hat{x}^*, 1$)	0.973	0.999	0.982	0.999
BAE, MC-Dropout	Ber(\hat{x}^*)	0.600	1.00	0.995	1.00
BAE, MC-Dropout	C-Ber(\hat{x}^*)	0.713	1.00	0.930	0.999
BAE, MC-Dropout	N($\hat{x}^*, 1$)	0.331	0.999	0.636	0.999
BAE, BayesBB	Ber(\hat{x}^*)	0.834	0.998	0.885	0.921
BAE, BayesBB	C-Ber(\hat{x}^*)	0.864	0.990	0.714	0.904
BAE, BayesBB	N($\hat{x}^*, 1$)	0.859	0.998	0.863	0.998
BAE, Ensemble	Ber(\hat{x}^*)	0.995	1.00	0.999	1.00
BAE, Ensemble	C-Ber(\hat{x}^*)	0.996	1.00	0.997	1.00
BAE, Ensemble	N($\hat{x}^*, 1$)	0.996	0.999	0.991	0.999

Table 4. (SVHN vs CIFAR10) AUROC for various models and likelihoods. Results where AUROC > 0.8 are in bold.

Model	Likelihood	AUROC			
		$\text{Var}_\theta(\hat{\mathbf{x}}^*)$	$\mathbb{E}_\theta(\text{LL})$	$\text{Var}_\theta(\text{LL})$	WAIC
Deterministic AE	Ber(\hat{x}^*)	-	0.442	-	-
Deterministic AE	C-Ber(\hat{x}^*)	-	0.436	-	-
Deterministic AE	N($\hat{x}^*, 1$)	-	0.972	-	-
VAE	Ber(\hat{x}^*)	0.751	0.455	0.948	0.456
VAE	C-Ber(\hat{x}^*)	0.526	0.435	0.947	0.436
VAE	N($\hat{x}^*, 1$)	0.630	0.973	0.964	0.973
BAE, MC-Dropout	Ber(\hat{x}^*)	0.820	0.445	0.923	0.446
BAE, MC-Dropout	C-Ber(\hat{x}^*)	0.861	0.441	0.898	0.445
BAE, MC-Dropout	N($\hat{x}^*, 1$)	0.802	0.967	0.938	0.967
BAE, BayesBB	Ber(\hat{x}^*)	0.649	0.570	0.797	0.606
BAE, BayesBB	C-Ber(\hat{x}^*)	0.855	0.485	0.877	0.502
BAE, BayesBB	N($\hat{x}^*, 1$)	0.760	0.918	0.851	0.915
BAE, Ensemble	Ber(\hat{x}^*)	0.922	0.440	0.950	0.441
BAE, Ensemble	C-Ber(\hat{x}^*)	0.921	0.436	0.939	0.439
BAE, Ensemble	N($\hat{x}^*, 1$)	0.931	0.969	0.960	0.969

⁵<https://pytorch.org/docs/stable/torchvision>

Table 5. (CIFAR10 vs SVHN) AUROC for various models and likelihoods. Results where AUROC>0.6 are in bold.

Model	Likelihood	AUROC			
		$\text{Var}_\theta(\hat{\mathbf{x}}^*)$	$\mathbb{E}_\theta(\text{LL})$	$\text{Var}_\theta(\text{LL})$	WAIC
Deterministic AE	Ber(\hat{x}^*)	-	0.601	-	-
Deterministic AE	C-Ber(\hat{x}^*)	-	0.607	-	-
Deterministic AE	N($\hat{x}^*, 1$)	-	0.038	-	-
VAE	Ber(\hat{x}^*)	0.492	0.604	0.051	0.604
VAE	C-Ber(\hat{x}^*)	0.518	0.609	0.077	0.609
VAE	N($\hat{x}^*, 1$)	0.522	0.045	0.048	0.045
BAE, MC-Dropout	Ber(\hat{x}^*)	0.241	0.600	0.113	0.599
BAE, MC-Dropout	C-Ber(\hat{x}^*)	0.214	0.607	0.144	0.605
BAE, MC-Dropout	N($\hat{x}^*, 1$)	0.260	0.051	0.088	0.051
BAE, BayesBB	Ber(\hat{x}^*)	0.377	0.572	0.211	0.570
BAE, BayesBB	C-Ber(\hat{x}^*)	0.385	0.591	0.193	0.586
BAE, BayesBB	N($\hat{x}^*, 1$)	0.359	0.094	0.131	0.094
BAE, Ensemble	Ber(\hat{x}^*)	0.186	0.603	0.086	0.603
BAE, Ensemble	C-Ber(\hat{x}^*)	0.175	0.609	0.104	0.609
BAE, Ensemble	N($\hat{x}^*, 1$)	0.151	0.039	0.065	0.039

C. Failure mode on CIFAR10 vs SVHN

We were unable to obtain any good results for this dataset pair (Table 5). In this scenario, we posit that the models are unable to learn meaningful representation of the training distribution. In turn, they have learnt to ‘copy’, as we examine the reconstructed images which are similar to the inputs, even on OOD dataset (Figure 4). When this occurs, even the $\text{Var}_\theta(\hat{\mathbf{x}}^*)$ and $\text{Var}(\text{LL})$ are unable to detect OOD inputs. Thus, as a cautionary note: although the AEs can reconstruct the in-distribution data with minimal error, they may not be immediately reliable for OOD detection. Even with sparsity and regularisation in the AEs, we were unable to circumvent this.

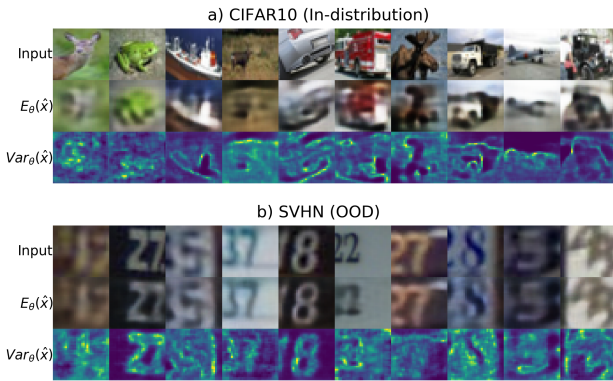


Figure 4. (CIFAR10 vs SVHN) Mean and uncertainty of reconstructed signals. In this failure mode, the reconstructed images of OOD data closely resemble the input images, which indicate the model has learnt to merely copy the inputs instead of learning the semantic representation of the training distribution.

D. Reconstructed images of test sets

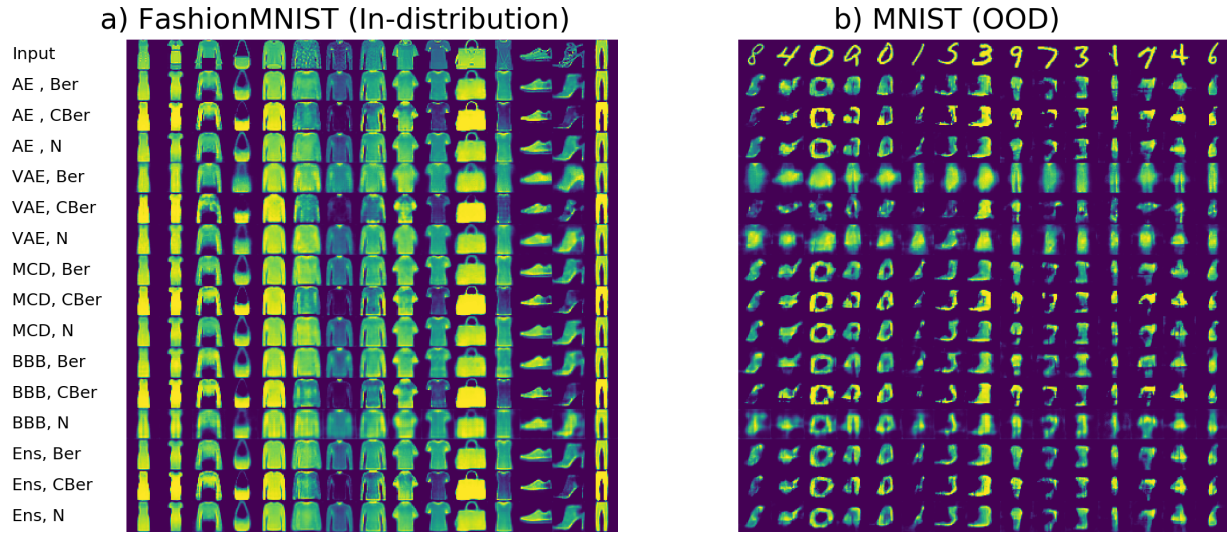


Figure 5. (FashionMNIST vs MNIST) Mean of reconstructed images for each model and likelihood.

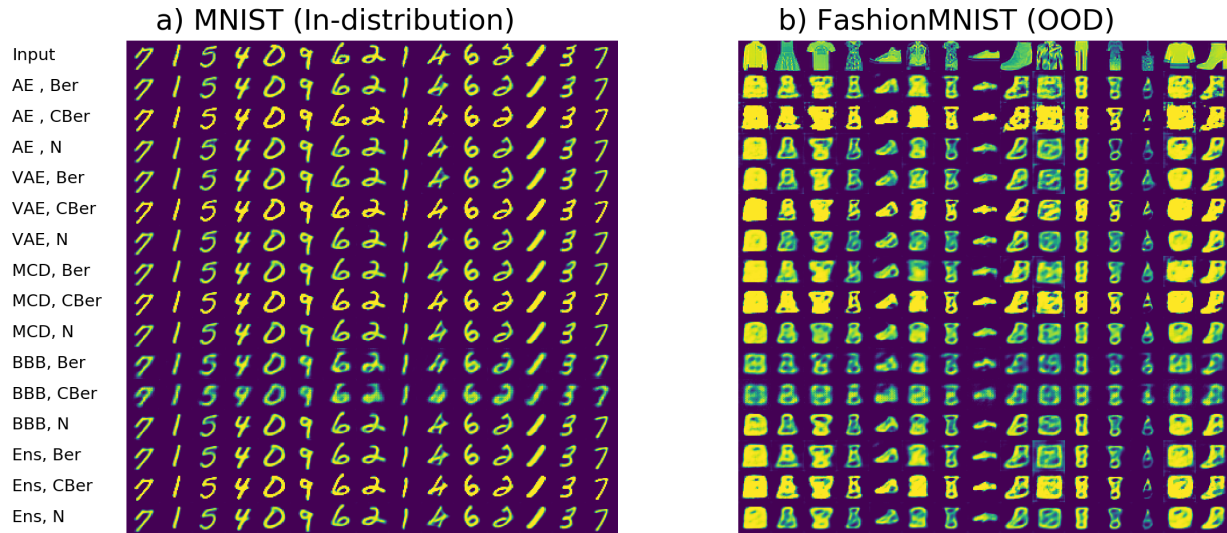


Figure 6. (MNIST vs FashionMNIST) Mean of reconstructed images for each model and likelihood.

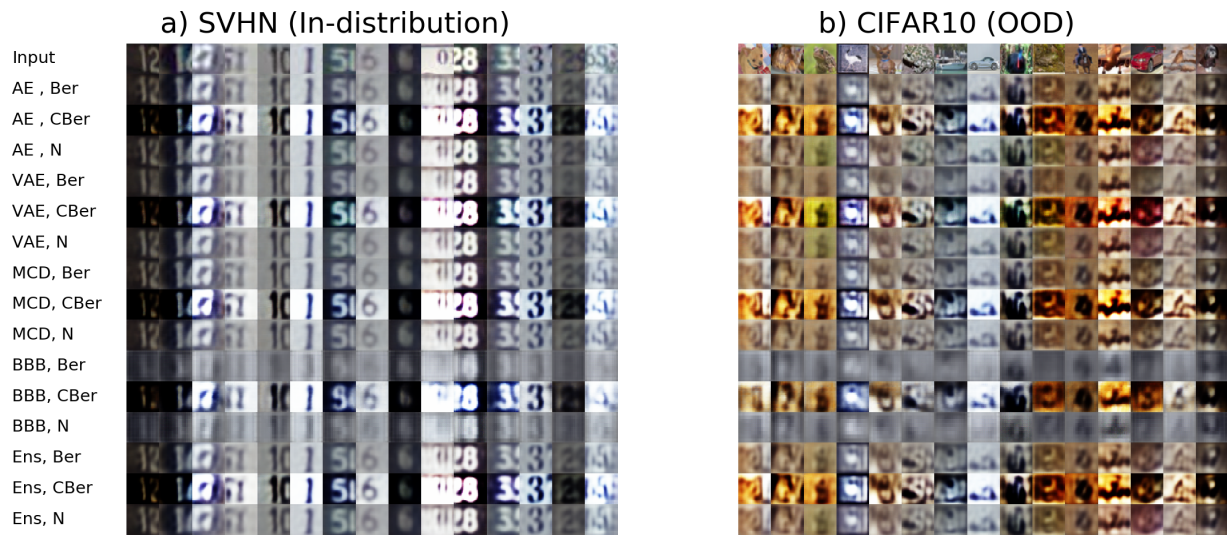


Figure 7. (SVHN vs CIFAR10) Mean of reconstructed images for each model and likelihood.

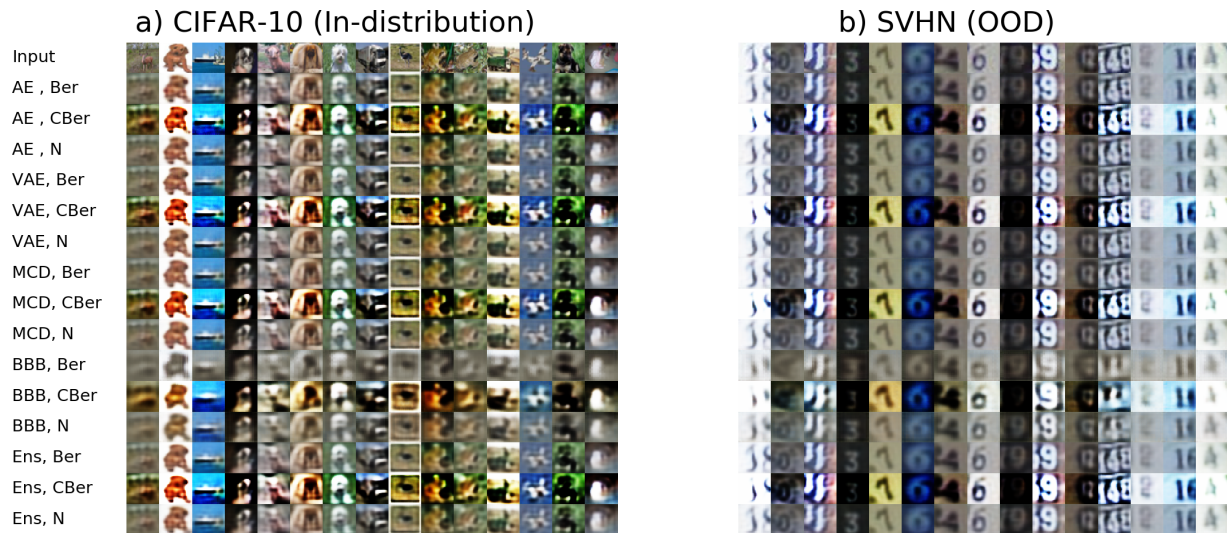


Figure 8. (CIFAR10 vs SVHN) Mean of reconstructed images for each model and likelihood.