ATML Group Report
Reproduced paper: Tighter Variational Bounds are Not Necessarily Better

M'Charrak, Amine
amine.mcharrak@cs.ox.ac.uk

Růžička, Vít
vit.ruzicka@cs.ox.ac.uk

Shin, Sangyun
sangyun.shin@cs.ox.ac.uk

Vankadari, Madhu
madhu.vankadari@cs.ox.ac.uk

April 2021

1 Introduction

Performing efficient approximate inference and learning for directed probabilistic models is a challenging problem as the latent variables of these models are continuous and have intractable posterior distributions. This problem can be solved by the Variational Bayes approach that involves an optimization step to approximate the intractable posterior. The common mean-field approaches cannot be applied to solve this problem as they demand the analytical solutions of the expected values w.r.t approximate posterior distributions which itself is intractable. In the recent works [9, 15], the authors proposed to solve this problem using a generative model which pairs a top-down generative network with a bottom-up recognition network. Both networks are jointly trained to maximize a variational lower bound on the intractable model evidence. These models are also known as Variational Autoencoders (VAE).

However, these models assume that the posterior is approximately factorial and that its parameters can be estimated from the input data using nonlinear regression. [3] argues that learning by maximizing a variational lower bound on the log-likelihood encourages the neural network to learn representations where the aforesaid assumptions are satisfied. Despite the benefits of these representations, the learned approximate posterior is not fully expressive because of the imposed constraints. Also, they argue that the existing VAE objective harshly penalizes approximate posterior samples which can not explain the data, even if the recognition network puts much of its probability mass on good explanations. To address this issue, they propose a generative model named importance weighted autoencoder (IWAE) that shares the VAE architecture but is trained on tighter variational bounds derived from importance weighting. In this method, the recognition network derives many samples (K) from the learned posterior and their weights are averaged with an intuition that as the number of samples K increases, the IWAE lower bound gets closer to true log-likelihood. Also, the importance sampling is expected to help the model to express the posterior that is not generally possible by vanilla VAE formulation.

In this report, we are going to explain the recent work [14] which argues that the aforesaid condition of having tighter evidence lower bounds (ELBO) for improved reconstruction performance is not necessarily better, and doing that deteriorates the overall performance of the model. This work has proven this fact with theoretical and empirical evidence, that increasing the number of importance samples in IWAE [3] degrades the signal-to-noise ratio (SNR) of the gradient estimator in the inference network and thereby affecting the full learning process. In other words, even though increasing K decreases the standard deviation of the gradients, it also reduces the magnitude of the true gradient faster, thereby increasing the relative variance of the gradient updates.

Extensive experiments are performed to understand the importance of K. These experiments suggest that tighter variational bounds are beneficial for the generative network and looser bounds are preferable for the inference network. With these insights, three methods are introduced namely: the partially importance weighted autoencoder (PIWAE), the multiply importance weighted autoencoder (MIWAE) and the combination importance weighted autoencoder (CIWAE). All these systems entail IWAE as a special case but use the importance weights in different ways to ensure a higher SNR of the gradient estimators. In our reproduction report the efficacy of these algorithms is tested on multiple datasets namely, MNIST, and Omniglot. By the end, we demonstrate that the new algorithms are able to approximate the posterior much closer to the true posterior than IWAE and also match the performance of the IWAE generative network or potentially outperforming it in the case of PIWAE.

In this report, we are replicating the original contributions of [14] and further add the following contributions:

1. CIWAE uses a convex combination ELBO losses from the IWAE and VAE with a constant β defining the importance of each of these ELBOs, which generally requires ablation study with various β values. We propose to solve this issue by defining the β as a learnable parameter with an initial value and let the stochastic optimization to find the best-possible value for the given task.

2. We studied the effect of increasing the number of parameters in the inference and generative network, to understand the effect of model complexity on the performance of the system.

3. The reproduced paper benchmarked their results with the MNIST dataset only. The authors do not report any of the cross dataset behaviours to understand the generalization aspect of their proposed algorithms. In this report, we perform the evaluation on another dataset named Omniglot along with MNIST.

4. Finally, we have released our implementations of these methods as an open-source Github repository[1].

The remainder of this report is organized as follows: The details of the methods proposed in the original paper [14] and our contributions are explained in Section 3. Various experimental studies with qualitative and quantitative results are delineated in Section 4. The conclusion remarks and the future scope of this work is presented in Section 5.

2 Related Work

This section details the previous approaches used to learn the deep generative models. Before the raise of VAE, some models were proposed with Botlzman distributions [19][17] with an advantage of modeling tractable and conditional distributions. However, it was impossible to sample data from these learned models or compute the partition function which hampered their progress [18]. On the other hand, a few models are defined using belief networks [12][7] which

[1] `https://github.com/madhubabuv/TightIWAE/`

are tractable and easy to sample from but, the conditional distributions become entwined because of the "explaining away effect". Later on, [5] proposed the wake-sleep algorithm using Helmholtz machines which approximates the posterior by learning a recognition network along with a generative network. Unfortunately, each network needs a different objective function during the training. Deep auto-regressive networks [7] solved this problem of two objectives by training their network using a single variational lower bound. In this similar line of research, [10] proposed another algorithm that reduces the stochasticity in the gradient updates using a third network to predict reward baselines (REINFORCE algorithm [21]) that minimizes the expected square of the learning signal during the training.

Variational autoencoders [8][15], as explained in the introduction section, are also generative models and also trained using the same objective functions same as [10]. Instead of using REINFORCE to reduce the variance in the gradient updates, they use a technique named "backprop through a random number generator" [21] for the reparametrization of the random choices. In VAEs, the modes are described with a simple distribution (Gaussian) followed by a deterministic mapping instead of a sequence of random choices.

There are also works by [11] and [1] that tried deriving log-probability lower bounds by avoiding recognition networks and perform the inference using importance sampling from the prior. In another work [6], a different variety of graphical inference models is proposed using importance weighing. Similar to IWAE [3], another model named Reweighted Wake-sleep [2] is another generative model that extends the original Wake-sleep algorithm with a recognition network. All these networks implicitly assume the tightening the variational bounds is good for both generative and recognition networks which is proven not to be necessary for [14].

3 Method

In order to explain the authors three newly proposed algorithms (MIWAE, PIWAE, and CIWAE) we first describe the basic concepts of the VAE methodology and its optimization objective, the ELBO. Moreover, we highlight how the IWAE, an extension of the VAE, conceptually differs from the VAE modeling objective. Finally, we provide technical insights into how each of the three proposed estimators address the diminishing SNR modeling issue of IWAEs.

3.1 VAE and evidence lower bound (ELBO)

The Variational Autoencoder (VAE) is a probabilistic deep learning model for density estimation and representation learning of continuous latent variable models $p_\theta(x|z)$ with intractable posterior distributions $p_\theta(z|x)$ over latent variables z given data observations x [13]. The distribution of interest arises by marginalization over the latent variables z

$$p(x) = \int p_\theta(x|z)p_\theta(z)\mathrm{d}z \tag{1}$$

with $p_\theta(z)$ being the prior distribution and often times set to be standard Normal distribution $\mathcal{N}(0, I)$. This marginal probability distribution (or likelihood) $p(x)$ is intractable when the latent variable space $\mathcal{Z}$ is high dimensional due to the expensive marginalization operation. As such, learning the parameters θ via maximum likelihood estimation (MLE) becomes infeasible.

Therefore, in order to avoid the exponential marginalization costs in Equation (1) the authors of [9] propose to instead optimize for the evidence lower-bound (ELBO) by introducing an inference network $q_\phi(z|x)$ with learnable parameters ϕ. This enables us to cast the inference problem as an optimization problem by learning an analytic approximation $q_\phi(z|x)$ of the true posterior distribution $p_\theta(z|x)$ which is intractable because of its dependence on the difficult to compute marginal likelihood

$$p_\theta(z|x) = \frac{p_\theta(x,z)}{p_\theta(x)}. \tag{2}$$

Given the fact, that the logarithm is a monotonically increasing function of its argument, one can instead infer the parameters θ by maximizing the log likelihood, $\log p(x)$, instead of the likelihood. We can rewrite Equation (1) by applying the logarithm of the likelihood. To make the integral with respect to z tractable, we can use the approximate posterior $q_\phi(z|x)$ instead of the intractable true posterior $p_\theta(z|x)$. With these adjustments we can now derive the tractable VAE evidence lower bound ($ELBO_{VAE}$) to the log likelihood as follows

$$\begin{aligned}
\log p(x) &= \log \int p_\theta(x|z)p_\theta(z)\mathrm{d}z \\
&= \log \int \frac{p_\theta(x|z)p_\theta(z)}{q_\phi(z|x)}\, q_\phi(z|x)\mathrm{d}z \\
&= \log \mathbb{E}_{q_\phi(z|x)}\left[\frac{p_\theta(x|z)p_\theta(z)}{q_\phi(z|x)}\right] \\
&\geq \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p_\theta(x|z)p_\theta(z)}{q_\phi(z|x)}\right] =: \mathcal{L}_{VAE}(= ELBO_{VAE}).
\end{aligned} \tag{3}$$

The inference problem is now an optimization problem because one needs to find a variational distribution $q_\phi(z|x)$ that maximizes $\mathcal{L}_{VAE}$. To make the computation tractable and the inference network expressive enough of the input data x, the approximate posterior is chosen to be factorial (e.g., a Gaussian distribution with diagonal covariance matrix) and its parameters ϕ are determined by a non-linear function of the data (e.g., a neural network). The gap between the evidence lower bound $\mathcal{L}_{VAE}$ and the log likelihood log$p(x)$ can be quantified by reformulating Equation 3 such that

$$\begin{aligned}
\mathcal{L}_{VAE} &= \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p_\theta(x|z)p_\theta(z)}{q_\phi(z|x)}\right] = \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p_\theta(z|x)p_\theta(x)}{q_\phi(z|x)}\right] \\
&= \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(x)\right] + \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{p_\theta(z|x)}{q_\phi(z|x)}\right] \\
&= \log p_\theta(x) + \mathbb{E}_{q_\phi(z|x)}\left[\log p_\theta(z|x) - \log q_\phi(z|x)\right] \\
&= \log p_\theta(x) - \mathbb{E}_{q_\phi(z|x)}\left[\log q_\phi(z|x) - \log p_\theta(z|x)\right] \\
&= \log p_\theta(x) - \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{q_\phi(z|x)}{p_\theta(z|x)}\right] \\
&= \log p_\theta(x) - D_{KL}\left(q_\phi(z|x)\,||\,p_\theta(z|x)\right) \qquad (4)
\end{aligned}$$

where we use the definition of the Kullback-Leibler (KL) divergence

$$D_{KL}(Q\,||\,P) = \mathbb{E}_Q\left[\log\left(\frac{Q}{P}\right)\right]$$

from the distribution P to the distribution Q for the last equivalence. From Equation 4 we observe that the difference between the marginal log likelihood log$p_\theta(x)$ and the ELBO ($\mathcal{L}_{VAE}$) is the KL divergence of the approximate posterior distribution $p_\phi(z|x)$ and the true posterior distribution $p_\theta(z|x)$. Given the non-negative KL divergence and the log likelihood of the data being a constant, one can maximize the ELBO by minimizing the KL divergence term between these two distributions. When the learned approximate posterior distribution of the inference network is identical to the true posterior distribution of the true latent variable model, then the KL divergence is zero and the ELBO equals the log likelihood of the data.

In practice a simple Monte Carlo estimator of the ELBO $\mathcal{L}_{VAE}$ can be used to approximate the expectation in Equation (3) by the sample mean such that

$$\hat{\mathcal{L}}_{VAE} = \frac{1}{L}\sum_{i=1}^{L} \log \frac{p_\theta(x|z_i)p_\theta(z_i)}{q_\phi(z_i|x)}. \qquad (5)$$

This estimator is unbiased because $\mathbb{E}_{z_i \sim q_\phi z|x}\left[\mathcal{L}_{VAE}\right] = \mathcal{L}$. Concurrent maximization of Equation (3) with respect to both neural networks' parameters θ, ϕ results in approximate maximum likelihood estimation of $p(x)$. This can be achieved through end-to-end learning using back propagation [16] and the reparameterisation trick [8] to allow for gradient optimisation methods despite the stochastic operations performed for retrieving samples z_i from the learned approximate posterior distribution over latent variables $q_\phi(z|x)$.

From an architectural perspective, the VAE fits an encoder, called inference network and typically parameterised as a neural network, to learn an arbitrary complex mapping between a data point x from the data space $\mathcal{X}$ and the latent variable z from the unobservable latent space $\mathcal{Z}$ with the distribution over unobserved variables $p_\theta(z|x)$. Using samples from the posterior distribution, a decoder, called generative network and also parameterised by a neural network, learns a mapping from the unobservable latent space $\mathcal{Z}$ back into the input space of our data observations $\mathcal{X}$. As such, the generative network enables the fitting of a conditional distribution $p_\theta(x|z)$. This distribution describes the generating process of the dataset $\mathcal{D} = \{x_i\}_{i=1}^N$, with each x_i being sampled i.i.d from the true generative process $p_\theta(x|z)$.

3.2 IWAE: A multiple sample VAE extension with tighter bounds

The VAE objective is limited in its ability to infer complex posterior distributions due to the strong assumptions it makes, e.g., a factorial true posterior distribution that can be learned with a feed-forward neural network [3]. For most real-world datasets these assumptions are oftentimes not met and hence the VAE objective results in learning an oversimplified approximate factorial posterior distribution. To alleviate these restrictions, [3] propose a tighter evidence lower bound, the IWAE ELBO, which employs a multiple (i.e., K) sample importance-sampling strategy on the marginal likelihood.

The IWAE objective can be derived by approximating the expectation inside the logarithm of Equation (3) before applying Jensen's inequality to retrieve the IWAE evidence lower bound. The authors employ Monte Carlo to estimate the expression inside the logarithm using the sample mean based on K samples from the approximate posterior distribution $q_\phi(z|x)$. Using the result from Equation (3), we can denote the $ELBO_{IWAE}$ as

$$\begin{aligned}
\log p(x) &= \log \mathbb{E}_{q_\phi(z|x)}\left[\frac{p_\theta(x|z)p_\theta(z)}{q_\phi(z|x)}\right] \\
&= \log\left(\mathbb{E}_{z_1,\dots,z_K\sim q_\phi(z|x)}\left[\frac{1}{K}\sum_{i=1}^{K}\frac{p_\theta(x|z_i)p_\theta(z_i)}{q_\phi(z_i|x)}\right]\right) \\
&\geq \mathbb{E}_{z_1,\dots,z_K\sim q_\phi(z|x)}\left[\log\left(\frac{1}{K}\sum_{i=1}^{K}\frac{p_\theta(x|z_i)p_\theta(z_i)}{q_\phi(z_i|x)}\right)\right] =: \mathcal{L}_{IWAE}^{K} (= ELBO_{IWAE}).
\end{aligned} \tag{6}$$

If we choose to optimize the IWAE ELBO in Equation (6) with $K = 1$, the VAE and IWAE ELBO estimators are equivalent. As K increases we get that $\mathcal{L}_{IWAE}^{K} \to \log p(x)$. Finally, for any $K \geq 1$ it can be shown that the importance-weighted autoencoder defines a variational inference strategy that allows for arbitrarily tight evidence lower bounds [4] such that

$$\log p(x) \geq \mathcal{L}_{IWAE}^{K+1} \geq \mathcal{L}_{IWAE}^{K} \geq \mathcal{L}_{IWAE}^{K=1} = \mathcal{L}_{VAE}.$$

The benefits of a tighter evidence lower bound for the IWAE comes with the cost of getting a biased estimator as compared to the VAE ELBO which enabled unbiased estimates of the marginal log likelihood [13]. Compared to the VAE with $K = 1$, setting $K > 1$ results in significant improvements for learning the generative model as measured by the marginal log likelihood of the data, $\log p(x)$.

Using Equation (6) the gradient estimator of the IWAE ELBO for the inference and generative network has the form

$$\Delta_K = \nabla_{\theta,\phi} \log \frac{1}{K} \sum_{k=1}^{K} w_k \tag{7}$$

where $w_k = \frac{p_\theta(x|z_k)p_\theta(z_k)}{q_\phi(z_k|x)}$ and $z_k \overset{i.i.d}{\sim} q_\phi(z_k|x)$.

3.3 Problem of tighter variational bounds

It was believed that increasing K ensures tighter variational bounds, leading to the better performance of VAE. However, the paper we chose [14] questions this, showing that increasing K doesn't always lead to better performance. To prove this, signal-to-noise ratio (SNR) is introduced for estimating the relative accuracy of gradient. SNR is defined as follows:

$$SNR_{M,K}(\theta) = |\,\mathbb{E}[\Delta_{M,K}(\theta)]/\sigma_{M,K}[(\theta)]| \tag{8}$$

where σ stands for the standard deviation of a random variable. The intuition behind SNR is that, although high SNR does not always mean that the gradient is well calculated without noise, low SNR is always detrimental due to the dominant noise in the estimates. To formally investigate this, [14] derives the convergence rates of SNR_ϕ and SNR_θ as follows:

$$SNR_{M,K}(\theta) = \sqrt{M} \left| \frac{\sqrt{K}\nabla_\theta Z - \frac{1}{2Z\sqrt{K}}\nabla_\theta \frac{Var[w_{1,1}]}{Z^2} + O(\frac{1}{K^{3/2}})}{\sqrt{\mathbb{E}[w_{1,1}^2(\nabla_\theta log w_{1,1} - \nabla_{theta} log Z)^2] + O(\frac{1}{K})}} \right| \tag{9}$$

$$SNR_{M,K}(\phi) = \sqrt{M} \left| \frac{\nabla_\phi Var[w_{1,1}] + O(\frac{1}{K})}{2Z\sqrt{K}\sigma[\nabla_\phi w_{1,1}] + O(\frac{1}{\sqrt{K}})} \right| \tag{10}$$

where $Z = p_\theta(x)$ is the true marginal likelihood.

Intuitively, according to $SNR_{M,K}(\theta)$ and $SNR_{M,K}(\phi)$, it is clear that increasing M always leads to higher $SNR_{M,K}(\theta)$ and $SNR_{M,K}(\phi)$, benefiting both inference and generative networks. However, when increasing K, SNR_θ becomes larger while SNR_ϕ becomes lower, suggesting that tighter variational bounds by increasing K is only good for the generative network and can be detrimental for the inference network. Based on this insight, [14] proposes three improved methods addressing the problem of the tighter variational bounds.

3.4 Multiply importance weighted auto-encoder (MIWAE)

From Equations 6 and 7 we observe that we are using only a single sample of $\log \frac{1}{K}\sum_{k=1}^{K} w_k$ to estimate importance weighted ELBO and its gradients with respect to the parameters of the inference and generative network.

As such, the first natural approach would be to reduce the variance of the gradient estimates $\nabla_{\theta,\phi}\mathcal{L}_{IWAE}^{K}$ by taking multiple samples, e.g., M, to estimate the gradients of the ELBO. As stated in [14], M does not affect the true value of the gradients of the lower bound. However, K, the number of samples used for the IWAE, does affect the bound by making it tighter.

The first of the three newly proposed algorithms proposed to address the issue of a vanishing SNR in the inference network is called MIWAE. This algorithm uses M instead of 1 sample in order to estimate $\mathcal{L}_{IWAE}^{K}$ and its gradients. The gradient estimates for the multiply importance weighted autoencoder are computed as follows:

$$\Delta_{M,K} = \frac{1}{M}\sum_{m=1}^{M}\left[\nabla_{\theta,\phi}\log\left(\frac{1}{K}\sum_{k=1}^{K} w_{m,k}\right)\right] \tag{11}$$

with $w_{m,k} = \frac{p_\theta(x|z_{m,k})p_\theta(z_k)}{q_\phi(z_{m,k}|x)} = \frac{p_\theta(z_{m,k},x)p_\theta(z_k)}{q_\phi(z_{m,k}|x)}$ and $z_{m,k} \overset{i.i.d}{\sim} q_\phi(z_{m,k}|x)$.

Assuming a fixed computation budget with $T = MK$ Monte Carlo samples, we can retrieve two special cases of the MIWAE namely:

- for $K = 1$ and $M = T$ we get a Monte Carlo estimator of the VAE lower bound objective as shown in Equation (5)
- for $K = T$ and $M = 1$ we get the corresponding Monte Carlo estimate for the IWAE lower bound objective.

The motivation for this approach is based on the observation that the SNR of the IWAE inference network increases with $O(\sqrt{M/K})$. As such, it should be possible the mitigate the detrimental effect of larger values of K by also increasing the number of samples, M, used for Monte Carlo estimation of the evidence lower bound in the IWAE. To be able to fairly compare the different algorithms proposed, the overall computation budget, T, will be held fixed for all proposed algorithms and evidence lower bounds.

3.5 Combination importance weighted auto-encoder (CIWAE

The second proposed algorithm, CIWAE, employs a convex combination of the variational lower bounds for the VAE and IWAE objective respectively. We have derived and discussed both evidence lower bounds, $\mathcal{L}_{VAE}$ and $\mathcal{L}_{IWAE}^{K}$, in the previous subsections. The evidence lower bound of the CIWAE takes the form

$$\mathcal{L}_{CIWAE}^{K} = \beta \mathcal{L}_{VAE} + (1-\beta)\mathcal{L}_{IWAE}^{K} \tag{12}$$

with $\beta \in [0,1]$ being a combination parameter that acts as a weighting on each of ELBO terms. Given that $\mathcal{L}_{CIWAE}^{K}$ is a combination of a the tighter IWAE bound and the looser VAE bound, it is trivial to see that

$$\mathcal{L}_{IWAE}^{K} \geq \mathcal{L}_{CIWAE}^{K} \geq \mathcal{L}_{VAE}.$$

Finally, the gradient estimator of the CIWAE algorithm takes on the following form

$$\Delta_{K,\beta}^{CIWAE} = \nabla_{\theta,\phi}\left(\beta \frac{1}{K}\sum_{k=1}^{K} \log w_k + (1-\beta) \log\left(\frac{1}{K}\sum_{k=1}^{K} w_k\right)\right). \tag{13}$$

The idea behind proposing the convex combination of two different ELBOs comes from the observations made for the SNR of the inference network. When the gradient estimates of the IWAE vanish, the non-zero gradient estimates of the VAE (even for small β) should enable the CIWAE modeling objective to alleviate SNR issues in the inference network (i.e., the gradient estimates with respect to ϕ). For the CIWAE algorithm, the SNR increases with $O(\sqrt{MK})$.

3.6 Partially importance weighted auto-encoder (PIWAE)

Finally, the last proposed algorithm is based on the insight, that an increased K in the IWAE objective is good for the generative network, however, detrimental to the learning signal in the inference network.

Therefore, the authors suggest to use different evidence lower bound targets for the inference network, $q_\phi(z|x)$, and the generative network, $p_\theta(x|z)$. To achieve this, the third and last proposed method, called partially IWAE, PIWAE, is introduced. The strategy of this variational inference method is to optimize the inference network based on the MIWAE target, allowing SNR adjustments by increasing the sample size M, and the generative network on the IWAE target, in order to benefit from the strictly tighter bound as K increases and thus leading to improved empirical gains in terms of the generative model's marginal likelihood estimation.

The corresponding PIWAE gradient estimates of the ELBO with respect to the inference network's parameters, ϕ, and the generative network's parameters, θ are given by

$$\Delta_{M,L}^{PIWAE}(\phi) = \frac{1}{M}\sum_{m=1}^{M}\left[\nabla_\phi \log\left(\frac{1}{L}\sum_{l=1}^{L} w_{m,l}\right)\right] \tag{14}$$

$$\Delta_K^{PIWAE}(\theta) = \nabla_\theta \log \frac{1}{K} \sum_{k=1}^{K} w_k \tag{15}$$

where L is the number of importance-weighted samples used for the evidence lower bound with regards to the optimization for parameters of the inference network. Instead of simultaneous optimization using a single target lower bound as in all other VAE and IWAE algorithms above, we now now have to optimize for two separate ELBO targets. This is achieved by using two separate optimizers, one which only modifies the parameters, ϕ of the inference network and another optimizer, which only affects the parameters, θ, of the generative network. In our implementation, we first update the encoding inference network and then the decoding generative network.

3.7 β as Learnable Parameter in CIWAE

Three proposed methods, such as MIWAE, CIWAE and PIWAE, are effectively alleviating the issue of SNR by different means. One of the possible improvements we found was to reduce the number of hyper-parameter that require cumbersome tuning. For instance, as was addressed in the paper, setting the "relatively low" β in CIWAE would help when the gradient of IWAE becomes very small. However, defining a "relatively small" value for the different dataset might not be a trivial task, requiring a large number of experiments. Also, in domains where the gradient of IWAE is always big enough, setting the small β would lead to suboptimal performance of the network. Therefore, it is natural to question if making the β learnable would be advantageous for VAE.

We use gradient descent for optimizing β since its optimization can be simultaneously performed with the optimization of VAE. Note that, however, optimizing β can be performed by any other optimization techniques. By defining $ELBO_{VAE}$ as $ELBO$ for VAE and $ELBO_{IWAE}$ as $ELBO$ using IWAE with K, the objective for optimizing β is:

$$\min_{\beta} \beta ELBO_{VAE} + (1-\beta) ELBO_{IWAE} \tag{16}$$

4 Results

4.1 Reproducing the result on MNIST and Omniglot

To reproduce the results of the original paper [14] we chose two benchmark datasets, the MNIST dataset of handwritten digits and the Omniglot dataset of diverse handwritten characters. We follow the dataset processing steps described in [3], and use the binarized versions of the two datasets alonside the official train and test data splits. Similarly, we chose the same model architectures as those described in [3], namely the model described as having a single stochastic layer. This model has two linear layers with a hidden dimension of 200 in both the Encoder and Decoder networks, and latent dimension of 50. As an

Table 1: Referential and tested larger model trained for 3280 epochs on the MNIST dataset. Numbers in the brackets indicate the (M, K) and the β.

Metric	IWAE	PIWAE$_{(8,8)}$	MIWAE$_{(8,8)}$	CIWAE$_{\beta=0.5}$	VAE
IWAE-64	-84.64	-84.72	-84.98	-87.05	-87.26
log$\hat{\text{p}}$(x)	-84.64	-84.90	-85.04	-87.00	-87.66
-KL(Q||P)	0.00	0.19	0.06	-0.05	0.40

Larger model	IWAE	PIWAE$_{(8,8)}$	MIWAE$_{(8,8)}$	CIWAE$_{\beta=0.5}$	VAE
IWAE-64	-83.85	-83.86	-83.47	-84.92	-85.33
log$\hat{\text{p}}$(x))	-83.92	-83.84	-83.66	-84.98	-85.31
-KL(Q||P)	0.06	-0.02	0.19	0.06	-0.02

exploration of these parameters, we chose to also train a larger model in terms of the number of parameters with the same layer layout and hidden dimension of 400 and latent dimension of 20 (note that these are also the basic parameters used in the official Pytorch example[2]). The number of trainable parameters in this proposed larger model is more than double when compared with the referential model (smaller model has 425284 and the larger model has 973624 parameters).

We have used the same experiment conditions when reproducing the results of the original paper [14]. Namely we train these models for 3280 epochs, use the Adam optimizer (with default parameters) and use the learning rate scheduling described in [3]. The final reproduction results are reported in Table 1 and the training progress can be in detail explored in Figure 1.

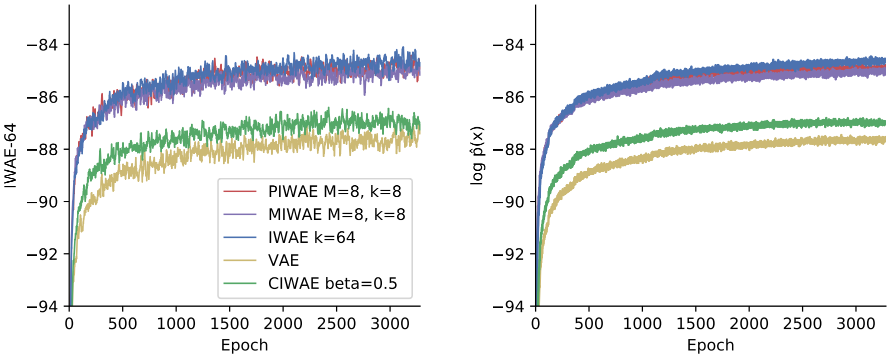
When evaluating the tested models on the Omniglot dataset, we lowered the number of epochs to 1000 due to limited available compute and differentiated learning plots as is shown in Figure 2. Exploratory results of the larger version of the model trained on MNIST can be seen in Figure 3.

Finally, we have also included the SSIM quantitative evaluation metric [20] used to compare visual quality between two images on all pairs of original and reconstructed images from the test set. Qualitative and quantitative comparison of the reconstructions of data from MNIST dataset is in Figure 5.

4.2 Experiment on model generalizability

Furthermore as a proxy measure for the generalizability of the trained models, we chose to evaluate models trained on one dataset and measure their SSIM score on another dataset. We note that while this experiment asks the models to perform a task which they were not trained on, due to the shared domain of datasets such as MNIST and Omniglot (handwritten digits and characters), we expect some overlapped knowledge in the representational power of the models. This can be used in zero-shot learning scenarios and as an evaluation of the expected quality of reused weights between two data domains which is commonly

[2] `https://github.com/pytorch/examples/blob/master/vae/main.py`



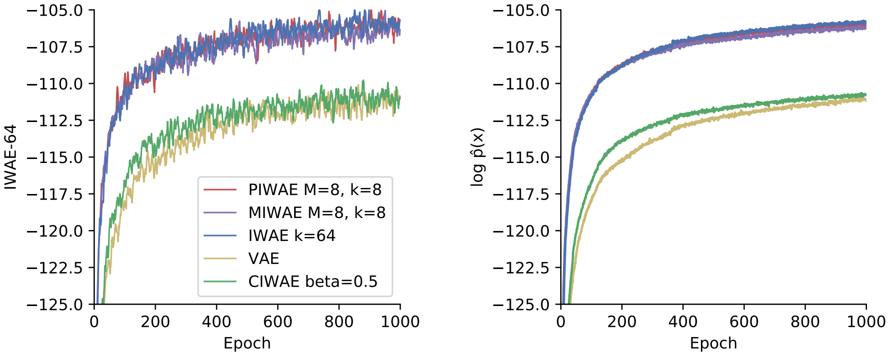
(a) IWAE_{64} (b) $\log \hat{\mathrm{p}}(\mathrm{x})$

Figure 1: The metrics of IWAE_{64} and $\log\hat{\mathrm{p}}(\mathrm{x})$ of referential models trained on the MNIST dataset across the training progress. We used rolling window of 10 for better clarity of the IWAE_{64} plot. Note that the figures are shared between (a) and (b).

used when fine-tuning models pre-trained on the ImageNet dataset. We show the quantitative evaluation of these results in Table 2 and show reconstructions in 7.

4.3 Experiment on learning β in CIWAE

The purpose of this experiment is to measure how learning beta in CIWAE affects system performance. Our main intuition behind this experiment is that the optimal value of β can be different depending on the dataset, and finding this value can be a cumbersome task. To see if different β leads to different result even in the same dataset, we first perform an experiment with different β. The following experiment is with learning β, to compare the performance with first experiment. Figure 4 shows the performance of the network for these two experiments in IWAE-64 metric. In the first experiment, when β is set differently, the performance of the network is estimated to have -81.29±2.98, indicating that different β can lead to a different result that can be suboptimal. On the other hand, when learning β, the average IWAE-64 metric is -80.34±0.49, suggesting that learning β helps minimize the deviation caused by the different initial value of β. Considering that learning β only adds one more optimization step using gradient descent, the gain by learning β for alleviating the parameter tuning step would be advantageous.



(a) IWAE_{64}

(b) $\log\hat{\mathrm{p}}(\mathrm{x})$

Figure 2: The metrics of IWAE_{64} and $\log\hat{\mathrm{p}}(\mathrm{x})$ of referential models trained on the Omniglot dataset across the training progress. We used rolling window of 5 in (a). Note that we have trained these models for 1000 epochs.

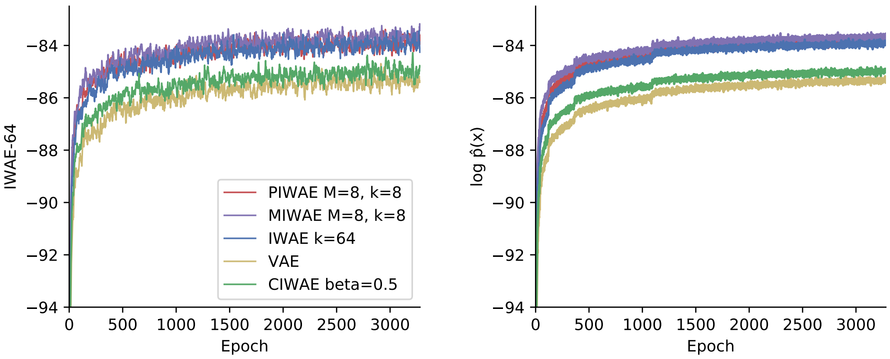
Table 2: Generalization ability of different models variants. Note that all models are being evaluated by SSIM on the Omniglot datasets, while only the second row was trained on it.

Train:	IWAE	$\mathrm{PIWAE}_{(8,8)}$	$\mathrm{MIWAE}_{(8,8)}$	$\mathrm{CIWAE}_{\beta=0.5}$	VAE
MNIST	0.58±0.11	**0.60±0.11**	0.59±0.12	0.58±0.12	0.57±0.12
Omniglot	0.72±0.09	0.74±0.08	**0.75±0.08**	0.72±0.10	0.72±0.10

5 Conclusion

In conclusion, we have reproduced the main methods proposed in [14] as an extension to the model of [3]. Furthermore, we have extended the evaluation of these methods over a new dataset (and tested the generalization ability of these models) and with altered basic model architecture (approximately doubling the number of parameters). Reproduction of the methods from [14] has yielded similar, but not exactly the same results, which might be due to implementation details or smaller scale of our experiments (namely only one run of the full 3280 epochs for each of the methods). Exploration of other hyper parameters, such as the choice of the architecture (dimensionality of the hidden and latent layers) has revealed larger performance gains than those achieved by varying the used learning method. We however note that these findings are orthogonal to each other and in practice we would strive to optimize both aspects - the size of the model architecture as well as the used method.

We also show that the β hyper parameter of CIWAE can be learned, and more importantly that this approach leads to higher robustness in terms of independence on the initialization of this parameter.



(a) IWAE_{64}

(b) $\log\hat{\mathrm{p}}(\mathrm{x})$

Figure 3: The metrics of IWAE_{64} and $\log\hat{\mathrm{p}}(\mathrm{x})$ on a larger version of models. We used rolling window of 10 in (a).

Evaluation of the reconstruction quality using the SSIM metric has indicated that the performance of the different models under this metric is comparable.

Lastly, our motivation to reproduce the methods of [14] was to confirm their findings on other datasets. To this effect we run experiments with the Omniglot dataset, where we could see similar behaviour as the one seen on the reproduced and referential results on the MNIST dataset. If we had more time and computational resources for this assignment, we would try to extend it to other real world scenario datasets (as can be also seen in the development branch of our Github repository). Additionally, we wanted to provide and open-source implementation of these proposed methods, as there is no official code repository for this paper. This has made our efforts slightly more complicated, namely due to the fact that small implementation details can lead to considerable differences in the training progress.

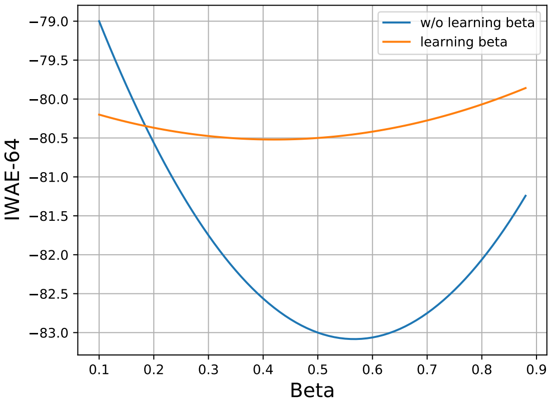
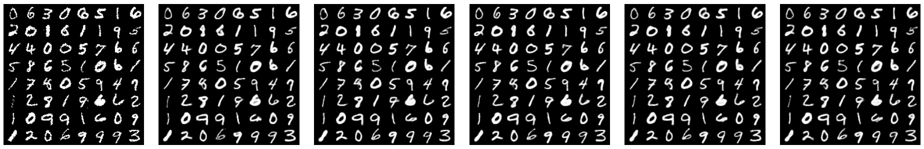


Figure 4: Illustration of IWAE-64 score with and without learning β. The average score when learning β is -80.34±0.49, whereas the average score is -81.99±2.98 without learning β.



(a) original input

(b) IWAE 0.816±0.051

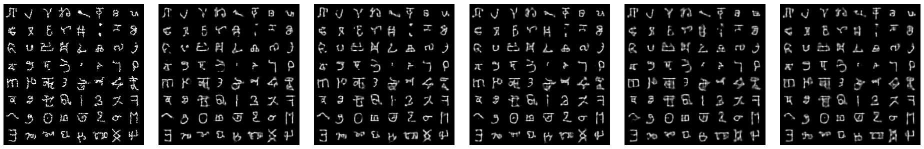
(c) PIWAE 0.832±0.047

(d) MIWAE **0.833±0.048**

(e) CIWAE **0.836±0.049**

(f) VAE 0.830±0.052

Figure 5: Qualitative and quantitative analysis of the reconstructed images on the MNIST dataset with the referential model size. We use the SSIM metric evaluated on the MNIST test set. All models were trained for 3280 epochs. Note (K, M) values are set to $(8, 8)$ and $\beta = 0.5$



(a) original input

(b) IWAE 0.724±0.086

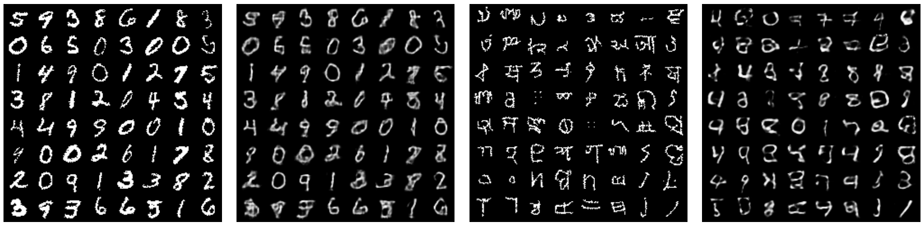
(c) PIWAE **0.743±0.082**

(d) MIWAE **0.745±0.082**

(e) CIWAE 0.719±0.098

(f) VAE 0.717±0.099

Figure 6: Qualitative and quantitative analysis of the reconstructed images on the Omniglot dataset with the referential model size. We use the SSIM metric evaluated on the Omniglot test set. All models were trained for 1000 epochs. Note (K, M) values are set to $(8, 8)$ and $\beta = 0.5$



(a) input MNIST

(b) reconstruction by a Omniglot trained model

(c) input Omniglot

(d) reconstruction by a MNIST trained model

Figure 7: Generalization ability measured by reconstruction quality between models trained on one dataset and evaluated on another dataset. In all cases we used the MIWAE$_{(8,8)}$ model for reconstruction.

Bibliography

[1] Jimmy Ba, Volodymyr Mnih, and Koray Kavukcuoglu. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*, 2014.

[2] Jörg Bornschein and Yoshua Bengio. Reweighted wake-sleep. *arXiv preprint arXiv:1406.2751*, 2014.

[3] Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*, 2015.

[4] C. Cremer, Q. Morris, and D. Duvenaud. Reinterpreting Importance-Weighted Autoencoders. *Workshop at the International Conference on Learning Representations*, 2017.

[5] Peter Dayan, Geoffrey E Hinton, Radford M Neal, and Richard S Zemel. The helmholtz machine. *Neural computation*, 7(5):889–904, 1995.

[6] Vibhav Gogate, Bozhena Bidyuk, and Rina Dechter. Studies in lower bounding probability of evidence using the markov inequality. 2007.

[7] Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. Deep autoregressive networks. In *International Conference on Machine Learning*, pages 1242–1250. PMLR, 2014.

[8] Diederik P Kingma, Tim Salimans, and Max Welling. Variational dropout and the local reparameterization trick. *arXiv preprint arXiv:1506.02557*, 2015.

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

[10] Andriy Mnih and Karol Gregor. Neural variational inference and learning in belief networks. In *International Conference on Machine Learning*, pages 1791–1799. PMLR, 2014.

[11] Radford M Neal. Learning stochastic feedforward networks. *Department of Computer Science, University of Toronto*, 64(1283):1577, 1990.

[12] Radford M Neal. Connectionist learning of belief networks. *Artificial intelligence*, 56(1):71–113, 1992.

[13] Sebastian Nowozin. Debiasing evidence approximations: On importance-weighted autoencoders and jackknife variational inference. In *ICLR 2018 Conference*, February 2018.

[14] Tom Rainforth, Adam Kosiorek, Tuan Anh Le, Chris Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. In *International Conference on Machine Learning*, pages 4277–4285. PMLR, 2018.

[15] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pages 1278–1286. PMLR, 2014.

[16] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. *Learning Internal Representations by Error Propagation*, page 318–362. MIT Press, Cambridge, MA, USA, 1986.

[17] Ruslan Salakhutdinov and Hugo Larochelle. Efficient learning of deep boltzmann machines. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 693–700. JMLR Workshop and Conference Proceedings, 2010.

[18] Ruslan Salakhutdinov and Iain Murray. On the quantitative analysis of deep belief networks. In *Proceedings of the 25th international conference on Machine learning*, pages 872–879, 2008.

[19] Paul Smolensky. Information processing in dynamical systems: Foundations of harmony theory. Technical report, Colorado Univ at Boulder Dept of Computer Science, 1986.

[20] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[21] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.