

---

# Tighter Variational Bounds are Not Necessarily Better

---

Tom Rainforth<sup>1</sup> Adam R. Kosiorek<sup>1,2</sup> Tuan Anh Le<sup>2</sup> Chris J. Maddison<sup>1</sup> Maximilian Igl<sup>2</sup> Frank Wood<sup>3</sup>  
Yee Whye Teh<sup>1</sup>

## Abstract

We provide theoretical and empirical evidence that using tighter evidence lower bounds (ELBOs) can be detrimental to the process of learning an inference network by reducing the signal-to-noise ratio of the gradient estimator. Our results call into question common implicit assumptions that tighter ELBOs are better variational objectives for simultaneous model learning and inference amortization schemes. Based on our insights, we introduce three new algorithms: the partially importance weighted auto-encoder (PIWAE), the multiply importance weighted auto-encoder (MIWAE), and the combination importance weighted auto-encoder (CIWAE), each of which includes the standard importance weighted auto-encoder (IWAE) as a special case. We show that each can deliver improvements over IWAE, even when performance is measured by the IWAE target itself. Furthermore, our results suggest that PIWAE may be able to deliver simultaneous improvements in the training of both the inference and generative networks.

## 1 Introduction

Variational bounds provide tractable and state-of-the-art objectives for training deep generative models (Kingma & Welling, 2014; Rezende et al., 2014). Typically taking the form of a lower bound on the intractable model evidence, they provide surrogate targets that are more amenable to optimization. In general, this optimization requires the generation of approximate posterior samples during the model training and so a number of methods simultaneously learn an *inference network* alongside the target *generative network*.

As well as assisting the training process, this inference network is often also of direct interest itself. For example, variational bounds are often used to train auto-encoders (Bourlard

& Kamp, 1988; Hinton & Zemel, 1994; Gregor et al., 2016; Chen et al., 2017), for which the inference network forms the encoder. Variational bounds are also used in amortized and traditional Bayesian inference contexts (Hoffman et al., 2013; Ranganath et al., 2014; Paige & Wood, 2016; Le et al., 2017), for which the generative model is fixed and the inference network is the primary target for the training.

The performance of variational approaches depends upon the choice of evidence lower bound (ELBO) and the formulation of the inference network, with the two often intricately linked to one another; if the inference network formulation is not sufficiently expressive, this can have a knock-on effect on the generative network (Burda et al., 2016). In choosing the ELBO, it is often implicitly assumed that using tighter ELBOs is universally beneficial, at least whenever this does not in turn lead to higher variance gradient estimates.

In this work we question this implicit assumption by demonstrating that, although using a tighter ELBO is typically beneficial to gradient updates of the generative network, it can be detrimental to updates of the inference network. Remarkably, we find that it is possible to simultaneously tighten the bound, reduce the variance of the gradient updates, and *arbitrarily deteriorate* the training of the inference network. Specifically, we present theoretical and empirical evidence that increasing the number of importance sampling particles,  $K$ , to tighten the bound in the importance-weighted auto-encoder (IWAE) (Burda et al., 2016), degrades the signal-to-noise ratio (SNR) of the gradient estimates for the inference network, inevitably deteriorating the overall learning process. In short, this behavior manifests because even though increasing  $K$  decreases the standard deviation of the gradient estimates, it decreases the magnitude of the true gradient faster, such that the *relative variance increases*.

Our results suggest that it may be best to use distinct objectives for learning the generative and inference networks, or that when using the same target, it should take into account the needs of both networks. Namely, while tighter bounds are typically better for training the generative network, looser bounds are often preferable for training the inference network. Based on these insights, we introduce three new algorithms: the partially importance-weighted auto-encoder (PIWAE), the multiply importance-weighted auto-encoder (MIWAE), and the combination importance-

---

<sup>1</sup>Department of Statistics, University of Oxford <sup>2</sup>Department of Engineering, University of Oxford <sup>3</sup>Department of Computer Science, University of British Columbia. Correspondence to: Tom Rainforth <rainforth@stats.ox.ac.uk>.

weighted auto-encoder (CIWAE). Each of these include IWAE as a special case and are based on the same set of importance weights, but use these weights in different ways to ensure a higher SNR for the inference network.

We demonstrate that our new algorithms can produce inference networks more closely representing the true posterior than IWAE, while matching the training of the generative network, or potentially even improving it in the case of PIWAE. Even when treating the IWAE objective itself as the measure of performance, all our algorithms are able to demonstrate clear improvements over IWAE.

## 2 Background and Notation

Let  $x$  be an  $\mathcal{X}$ -valued random variable defined via a process involving an unobserved  $\mathcal{Z}$ -valued random variable  $z$  with joint density  $p_\theta(x, z)$ . Direct maximum likelihood estimation of  $\theta$  is generally intractable if  $p_\theta(x, z)$  is a deep generative model due to the marginalization of  $z$ . A common strategy is to instead optimize a variational lower bound on  $\log p_\theta(x)$ , defined via an auxiliary inference model  $q_\phi(z|x)$ :

$$\begin{aligned} \text{ELBO}_{\text{VAE}}(\theta, \phi, x) &:= \int q_\phi(z|x) \log \frac{p_\theta(x, z)}{q_\phi(z|x)} dz \\ &= \log p_\theta(x) - \text{KL}(q_\phi(z|x) || p_\theta(z|x)). \end{aligned} \quad (1)$$

Typically,  $q_\phi$  is parameterized by a neural network, for which the approach is known as the variational auto-encoder (VAE) (Kingma & Welling, 2014; Rezende et al., 2014). Optimization is performed with stochastic gradient ascent (SGA) using unbiased estimates of  $\nabla_{\theta, \phi} \text{ELBO}_{\text{VAE}}(\theta, \phi, x)$ . If  $q_\phi$  is reparameterizable, then given a reparameterized sample  $z \sim q_\phi(z|x)$ , the gradients  $\nabla_{\theta, \phi} (\log p_\theta(x, z) - \log q_\phi(z|x))$  can be used for the optimization.

The VAE objective places a harsh penalty on mismatch between  $q_\phi(z|x)$  and  $p_\theta(z|x)$ ; optimizing jointly in  $\theta, \phi$  can confound improvements in  $\log p_\theta(x)$  with reductions in the KL (Turner & Sahani, 2011). Thus, research has looked to develop bounds that separate the tightness of the bound from the expressiveness of the class of  $q_\phi$ . For example, the IWAE objectives (Burda et al., 2016), which we denote as  $\text{ELBO}_{\text{IS}}(\theta, \phi, x)$ , are a family of bounds defined by

$$\begin{aligned} Q_{\text{IS}}(z_{1:K}|x) &:= \prod_{k=1}^K q_\phi(z_k|x), \\ \hat{Z}_{\text{IS}}(z_{1:K}, x) &:= \frac{1}{K} \sum_{k=1}^K \frac{p_\theta(x, z_k)}{q_\phi(z_k|x)}, \end{aligned} \quad (2)$$

$\text{ELBO}_{\text{IS}}(\theta, \phi, x) := \int Q_{\text{IS}}(z_{1:K}|x) \log \hat{Z}_{\text{IS}}(z_{1:K}, x) dz_{1:K} \leq \log p_\theta(x)$ . The IWAE objectives generalize the VAE objective ( $K = 1$  corresponds to the VAE) and the bounds become strictly tighter as  $K$  increases (Burda et al., 2016). When the family of  $q_\phi$  contains the true posteriors, the global optimum parameters  $\{\theta^*, \phi^*\}$  are independent of  $K$ , see e.g. (Le et al., 2018). Nonetheless, except for the most trivial models, it is not usually the case that  $q_\phi$  contains the true posteriors, and Burda et al. (2016) provide strong empirical evidence that

setting  $K > 1$  leads to significant empirical gains over the VAE in terms of learning the generative model.

Optimizing tighter bounds is usually empirically associated with better models  $p_\theta$  in terms of marginal likelihood on held out data. Other related approaches extend this to sequential Monte Carlo (SMC) (Maddison et al., 2017; Le et al., 2018; Naesseth et al., 2018) or change the lower bound that is optimized to reduce the bias (Li & Turner, 2016; Bamler et al., 2017). A second, unrelated, approach is to tighten the bound by improving the expressiveness of  $q_\phi$  (Salimans et al., 2015; Tran et al., 2015; Rezende & Mohamed, 2015; Kingma et al., 2016; Maaløe et al., 2016; Ranganath et al., 2016). In this work, we focus on the former, algorithmic, approaches to tightening bounds.

## 3 Assessing the Signal-to-Noise Ratio of the Gradient Estimators

Because it is not feasible to analytically optimize any ELBO in complex models, the effectiveness of any particular choice of ELBO is linked to our ability to numerically solve the resulting optimization problem. This motivates us to examine the effect  $K$  has on the variance and magnitude of the gradient estimates of IWAE for the two networks. More generally, we study IWAE gradient estimators constructed as the average of  $M$  estimates, each built from  $K$  independent particles. We present a result characterizing the asymptotic signal-to-noise ratio in  $M$  and  $K$ . For the standard case of  $M = 1$ , our result shows that the signal-to-noise ratio of the reparameterization gradients of the inference network for the IWAE decreases with rate  $O(1/\sqrt{K})$ .

As estimating the ELBO requires a Monte Carlo estimation of an expectation over  $z$ , we have two sample sizes to tune for the estimate: the number of samples  $M$  used for Monte Carlo estimation of the ELBO and the number of importance samples  $K$  used in the bound construction. Here  $M$  does not change the true value of  $\nabla_{\theta, \phi} \text{ELBO}$ , only our variance in estimating it, while changing  $K$  changes the ELBO itself, with larger  $K$  leading to tighter bounds (Burda et al., 2016). Presuming that reparameterization is possible, we can express our gradient estimate in the general form

$$\Delta_{M,K} := \frac{1}{M} \sum_{m=1}^M \nabla_{\theta, \phi} \log \frac{1}{K} \sum_{k=1}^K w_{m,k}, \quad (3)$$

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

Thus, for a fixed budget of  $T = MK$  samples, we have a family of estimators with the cases  $K = 1$  and  $M = 1$  corresponding respectively to the VAE and IWAE objectives. We will use  $\Delta_{M,K}(\theta)$  to refer to gradient estimates with respect to  $\theta$  and  $\Delta_{M,K}(\phi)$  for those with respect to  $\phi$ .

Variance is not always a good barometer for the effectiveness of a gradient estimation scheme; estimators with small expected values need proportionally smaller variances to be estimated accurately. In the case of IWAE, when changes in  $K$

simultaneously affect both the variance and expected value, the quality of the estimator for learning can actually *worsen* as the variance decreases. To see why, consider the marginal likelihood estimates  $\hat{Z}_{m,K} = \sum_{k=1}^K w_{m,k}$ . Because these become exact (and thus independent of the proposal) as  $K \rightarrow \infty$ , it must be the case that  $\lim_{K \rightarrow \infty} \Delta_{M,K}(\phi) = 0$ . Thus as  $K$  becomes large, the expected value of the gradient must decrease along with its variance, such that the variance relative to the problem scaling need not actually improve.

To investigate this formally, we introduce the signal-to-noise-ratio (SNR), defining it to be the absolute value of the expected estimate scaled by its standard deviation:

$$\text{SNR}_{M,K}(\theta) = |\mathbb{E}[\Delta_{M,K}(\theta)] / \sigma[\Delta_{M,K}(\theta)]| \quad (4)$$

where  $\sigma[\cdot]$  denotes the standard deviation of a random variable. The SNR is defined separately on each dimension of the parameter vector and similarly for  $\text{SNR}_{M,K}(\phi)$ . It provides a measure of the relative accuracy of the gradient estimates. Though a high SNR does not always indicate a good SGA scheme (as the target objective itself might be poorly chosen), a low SNR is always problematic as it indicates that the gradient estimates are dominated by noise: if  $\text{SNR} \rightarrow 0$  then the estimates become completely random. We are now ready to state our main theoretical result:  $\text{SNR}_{M,K}(\theta) = O(\sqrt{MK})$  and  $\text{SNR}_{M,K}(\phi) = O(\sqrt{M/K})$ .

**Theorem 1.** *Assume that when  $M = K = 1$ , the expected gradients; the variances of the gradients; and the first four moments of  $w_{1,1}$ ,  $\nabla_{\theta} w_{1,1}$ , and  $\nabla_{\phi} w_{1,1}$  are all finite and the variances are also non-zero. Then the signal-to-noise ratios of the gradient estimates converge at the following rates*

$$\text{SNR}_{M,K}(\theta) = \quad (5)$$

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

$$\text{SNR}_{M,K}(\phi) = \sqrt{M} \left| \frac{\nabla_{\phi} \text{Var}[w_{1,1}] + O\left(\frac{1}{K}\right)}{2Z\sqrt{K} \sigma[\nabla_{\phi} w_{1,1}] + O\left(\frac{1}{\sqrt{K}}\right)} \right| \quad (6)$$

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

*Proof.* We give an intuitive demonstration of the result here and provide a formal proof in Appendix A. The effect of  $M$  on the SNR follows from using the law of large numbers on the random variable  $\nabla_{\theta,\phi} \log \hat{Z}_{m,K}$ . Namely, the overall expectation is independent of  $M$  and the variance reduces at a rate  $O(1/M)$ . The effect of  $K$  is more complicated but is perhaps most easily seen by noting that (Burda et al., 2016)

$$\nabla_{\theta,\phi} \log \hat{Z}_{m,K} = \sum_{k=1}^K \frac{w_{m,k}}{\sum_{\ell=1}^K w_{m,\ell}} \nabla_{\theta,\phi} \log(w_{m,k}),$$

such that  $\nabla_{\theta,\phi} \log \hat{Z}_{m,K}$  can be interpreted as a self-normalized importance sampling estimate. We can, there-

fore, invoke the known result (see e.g. Hesterberg (1988)) that the bias of a self-normalized importance sampler converges at a rate  $O(1/K)$  and the standard deviation at a rate  $O(1/\sqrt{K})$ . We thus see that the SNR converges at a rate  $O((1/K)/(1/\sqrt{K})) = O(1/\sqrt{K})$  if the asymptotic gradient is 0 and  $O((1)/(1/\sqrt{K})) = O(\sqrt{K})$  otherwise, giving the convergence rates in the  $\phi$  and  $\theta$  cases respectively.  $\square$

The implication of these rates is that increasing  $M$  is monotonically beneficial to the SNR for both  $\theta$  and  $\phi$ , but that increasing  $K$  is beneficial to the former and detrimental to the latter. We emphasize that this means the SNR for the IWAE inference network gets worse as we increase  $K$ : this is not just an opportunity cost from the fact that we could have increased  $M$  instead, increasing the total number of samples used in the estimator actually worsens the SNR!

### 3.1 Asymptotic Direction

An important point of note is that the dependence of the true inference network gradients becomes independent of  $K$  as  $K$  becomes large. Namely, because we have as an intermediary result from deriving the SNRs that

$$\mathbb{E}[\Delta_{M,K}(\phi)] = -\frac{\nabla_{\phi} \text{Var}[w_{1,1}]}{2KZ^2} + O\left(\frac{1}{K^2}\right), \quad (7)$$

we see that expected gradient points in the direction of  $-\nabla_{\phi} \text{Var}[w_{1,1}]$  as  $K \rightarrow \infty$ . This direction is rather interesting: it implies that as  $K \rightarrow \infty$ , the optimal  $\phi$  is that which minimizes the variance of the weights. This is well known to be the optimal importance sampling distribution in terms of estimating the marginal likelihood (Owen, 2013). Given that the role of the inference network during training is to estimate the marginal likelihood, this is thus arguably exactly what we want to optimize for. As such, this result, which complements those of (Cremer et al., 2017), suggests that increasing  $K$  provides a preferable target in terms of the direction of the true inference network gradients. We thus see that there is a trade-off with the fact that increasing  $K$  also diminishes the SNR, reducing the estimates to pure noise if  $K$  is set too high. In the absence of other factors, there may thus be a ‘‘sweet-spot’’ for setting  $K$ .

### 3.2 Multiple Data Points

Typically when training deep generative models, one does not optimize a single ELBO but instead its average over multiple data points, i.e.

$$\mathcal{J}(\theta, \phi) := \frac{1}{N} \sum_{n=1}^N \text{ELBO}_{\text{IS}}(\theta, \phi, x^{(n)}). \quad (8)$$

Our results extend to this setting because the  $z$  are drawn independently for each  $x^{(n)}$ , so

$$\mathbb{E}\left[\frac{1}{N} \sum_{n=1}^N \Delta_{M,K}^{(n)}\right] = \frac{1}{N} \sum_{n=1}^N \mathbb{E}\left[\Delta_{M,K}^{(n)}\right], \quad (9)$$

$$\text{Var}\left[\frac{1}{N} \sum_{n=1}^N \Delta_{M,K}^{(n)}\right] = \frac{1}{N^2} \sum_{n=1}^N \text{Var}\left[\Delta_{M,K}^{(n)}\right]. \quad (10)$$

We thus also see that if we are using mini-batches such that  $N$  is a chosen parameter and the  $x^{(n)}$  are drawn from the

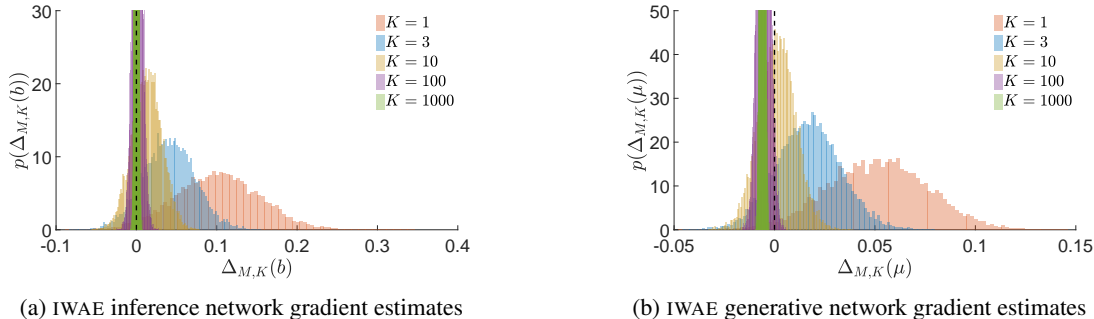


Figure 1: Histograms of gradient estimates  $\Delta_{M,K}$  for the generative network and the inference network using the IWAE ( $M = 1$ ) objective with different values of  $K$ .

empirical data distribution, then the SNRs of  $\bar{\Delta}_{N,M,K} := \frac{1}{N} \sum_{n=1}^N \Delta_{M,K}^{(n)}$  scales as  $\sqrt{N}$ , i.e.  $\text{SNR}_{N,M,K}(\theta) = O(\sqrt{NMK})$  and  $\text{SNR}_{N,M,K}(\phi) = O(\sqrt{NM/K})$ . Therefore increasing  $N$  has the same ubiquitous benefit as increasing  $M$ . In the rest of the paper, we will implicitly be considering the SNRs for  $\bar{\Delta}_{N,M,K}$ , but will omit the dependency on  $N$  to simplify the notation.

#### 4 Empirical Confirmation

Our convergence results hold exactly in relation to  $M$  (and  $N$ ) but are only asymptotic in  $K$  due to the higher order terms. Therefore their applicability should be viewed with a healthy degree of skepticism in the small  $K$  regime. With this in mind, we now present empirical support for our theoretical results and test how well they hold in the small  $K$  regime using a simple Gaussian model, for which we can analytically calculate the ground truth.

Consider a family of generative models with  $\mathbb{R}^D$ -valued latent variables  $z$  and observed variables  $x$ :

$$z \sim \mathcal{N}(z; \mu, I), \quad x|z \sim \mathcal{N}(x; z, I), \quad (11)$$

which is parameterized by  $\theta := \mu$ . Let the inference network be parameterized by  $\phi = (A, b)$ ,  $A \in \mathbb{R}^{D \times D}$ ,  $b \in \mathbb{R}^D$  where  $q_\phi(z|x) = \mathcal{N}(z; Ax + b, \frac{2}{3}I)$ . Given a dataset  $(x^{(n)})_{n=1}^N$ , we can analytically calculate the optimum of our target  $\mathcal{J}(\theta, \phi)$  as explained in Appendix B, giving  $\theta^* := \mu^* = \frac{1}{N} \sum_{n=1}^N x^{(n)}$  and  $\phi^* := (A^*, b^*)$ , where  $A^* = I/2$  and  $b^* = \mu^*/2$ . Though this will not be the case in general, for this particular problem, the optimal proposal is independent of  $K$ . However, the expected gradients for the inference network still change with  $K$ .

To conduct our investigation, we randomly generated a synthetic dataset from the model with  $D = 20$  dimensions,  $N = 1024$  data points, and a true model parameter value  $\mu_{\text{true}}$  that was itself randomly generated from a unit Gaussian, i.e.  $\mu_{\text{true}} \sim \mathcal{N}(\mu_{\text{true}}; 0, I)$ . We then considered the gradient at a random point in the parameter space close to optimum (we also consider a point far from the optimum in Appendix C.3). Namely each dimension of each parameter was randomly offset from its optimum value using a zero-mean Gaussian with standard deviation 0.01. We then

calculated empirical estimates of the ELBO gradients for IWAE, where  $M = 1$  is held fixed and we increase  $K$ , and for VAE, where  $K = 1$  is held fixed and we increase  $M$ . In all cases we calculated  $10^4$  such estimates and used these samples to provide empirical estimates for, amongst other things, the mean and standard deviation of the estimator, and thereby an empirical estimate for the SNR.

We start by examining the qualitative behavior of the different gradient estimators as  $K$  increases as shown in Figure 1. This shows histograms of the IWAE gradient estimators for a single parameter of the inference network (left) and generative network (right). We first see in Figure 1a that as  $K$  increases, both the magnitude and the standard deviation of the estimator decrease for the inference network, with the former decreasing faster. This matches the qualitative behavior of our theoretical result, with the SNR ratio diminishing as  $K$  increases. In particular, the probability that the gradient is positive or negative becomes roughly equal for larger values of  $K$ , meaning the optimizer is equally likely to increase as decrease the inference network parameters at the next iteration. By contrast, for the generative network, IWAE converges towards a non-zero gradient, such that, even though the SNR initially decreases with  $K$ , it then rises again, with a very clear gradient signal for  $K = 1000$ .

To provide a more rigorous analysis, we next directly examine the convergence of the SNR. Figure 2 shows the convergence of the estimators with increasing  $M$  and  $K$ . The observed rates for the inference network (Figure 2a) correspond to our theoretical results, with the suggested rates observed all the way back to  $K = M = 1$ . As expected, we see that as  $M$  increases, so does  $\text{SNR}_{M,K}(b)$ , but as  $K$  increases,  $\text{SNR}_{M,K}(b)$  reduces.

In Figure 2b, we see that the theoretical convergence for  $\text{SNR}_{M,K}(\mu)$  is again observed exactly for variations in  $M$ , but a more unusual behavior is seen for variations in  $K$ , where the SNR initially decreases before starting to increase again for large enough  $K$ , eventually exhibiting behavior consistent with the theoretical result for large enough  $K$ . The driving factor for this is that here  $\mathbb{E}[\Delta_{M,\infty}(\mu)]$  has a smaller magnitude than (and opposite sign to)  $\mathbb{E}[\Delta_{M,1}(\mu)]$

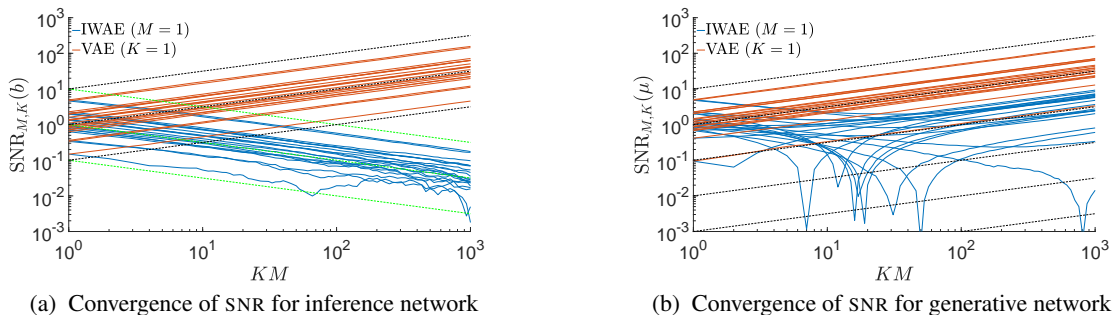


Figure 2: Convergence of signal-to-noise ratios of gradient estimates with increasing  $M$  and  $K$ . Different lines correspond to different dimensions of the parameter vectors. Shown in blue is the IWAE where we keep  $M = 1$  fixed and increase  $K$ . Shown in red is the VAE where  $K = 1$  is fixed and we increase  $M$ . The black and green dashed lines show the expected convergence rates from our theoretical results, representing gradients of  $1/2$  and  $-1/2$  respectively.

(see Figure 1b). If we think of the estimators for all values of  $K$  as biased estimates for  $\mathbb{E}[\Delta_{M,\infty}(\mu)]$ , we see from our theoretical results that this bias decreases faster than the standard deviation. Consequently, while the magnitude of this bias remains large compared to  $\mathbb{E}[\Delta_{M,\infty}(\mu)]$ , it is the predominant component in the true gradient and we see similar SNR behavior as in the inference network.

Note that this does not mean that the estimates are getting worse for the generative network. As we increase  $K$  our bound is getting tighter and our estimates closer to the true gradient for the target that we actually want to optimize  $\nabla_{\mu} \log Z$ . See Appendix C.2 for more details. As we previously discussed, it is also the case that increasing  $K$  could be beneficial for the inference network even if it reduces the SNR by improving the direction of the expected gradient. However, as we will now show, the SNR is, for this problem, the dominant effect for the inference network.

#### 4.1 Directional Signal-to-Noise Ratio

As a reassurance that our chosen definition of the SNR is appropriate for the problem at hand and to examine the effect of multiple dimensions explicitly, we now also consider an alternative definition of the SNR that is similar (though distinct) to that used in (Roberts & Tedrake, 2009). We refer to this as the “directional” SNR (DSNR). At a high-level, we define the DSNR by splitting each gradient estimate into two component vectors, one parallel to the true gradient and one perpendicular, then taking the expectation of ratio of their magnitudes. More precisely, we define  $u = \mathbb{E}[\Delta_{M,K}] / \|\mathbb{E}[\Delta_{M,K}]\|_2$  as being the true normalized gradient direction and then the DSNR as

$$\text{DSNR}_{M,K} = \mathbb{E} \left[ \frac{\|\Delta_{\parallel}\|_2}{\|\Delta_{\perp}\|_2} \right] \quad \text{where} \quad (12)$$

$$\Delta_{\parallel} = (\Delta_{M,K}^T u) u \quad \text{and} \quad \Delta_{\perp} = \Delta_{M,K} - \Delta_{\parallel}.$$

The DSNR thus provides a measure of the expected proportion of the gradient that will point in the true direction. For perfect estimates of the gradients, then  $\text{DSNR} \rightarrow \infty$ , but unlike the SNR, arbitrarily bad estimates do not have

$\text{DSNR} = 0$  because even random vectors will have a component of their gradient in the true direction.

The convergence of the DSNR is shown in Figure 3, for which the true normalized gradient  $u$  has been estimated empirically, noting that this varies with  $K$ . We see a similar qualitative behavior to the SNR, with the gradients of IWAE for the inference network degrading to having the same directional accuracy as drawing a random vector. Interestingly, the DSNR seems to be following the same asymptotic convergence behavior as the SNR for both networks in  $M$  (as shown by the dashed lines), even though we have no theoretical result to suggest this should occur.

As our theoretical results suggest that the direction of the true gradients correspond to targeting an improved objective as  $K$  increases, we now examine whether this or the changes in the SNR is the dominant effect. To this end, we repeat our calculations for the DSNR but take  $u$  as the target direction of the gradient for  $K = 1000$ . This provides a measure of how varying  $M$  and  $K$  affects the quality of the gradient directions as biased estimators for  $\mathbb{E}[\Delta_{1,1000}] / \|\mathbb{E}[\Delta_{1,1000}]\|_2$ . As shown in Figure 4, increasing  $K$  is still detrimental for the inference network by this metric, even though it brings the expected gradient estimate closer to the target gradient. By contrast, increasing  $K$  is now monotonically beneficial for the generative network. Increasing  $M$  leads to initial improvements for the inference network before plateauing due to the bias of the estimator. For the generative network, increasing  $M$  has little impact, with the bias being the dominant factor throughout. Though this metric is not an absolute measure of performance of the SGA scheme, e.g. because high bias may be more detrimental than high variance, it is nonetheless a powerful result in suggesting that increasing  $K$  can be detrimental to learning the inference network.

## 5 New Estimators

Based on our theoretical results, we now introduce three new algorithms that address the issue of diminishing SNR for the inference network. Our first, MIWAE, is exactly equivalent to the general formulation given in (3), the distinction from

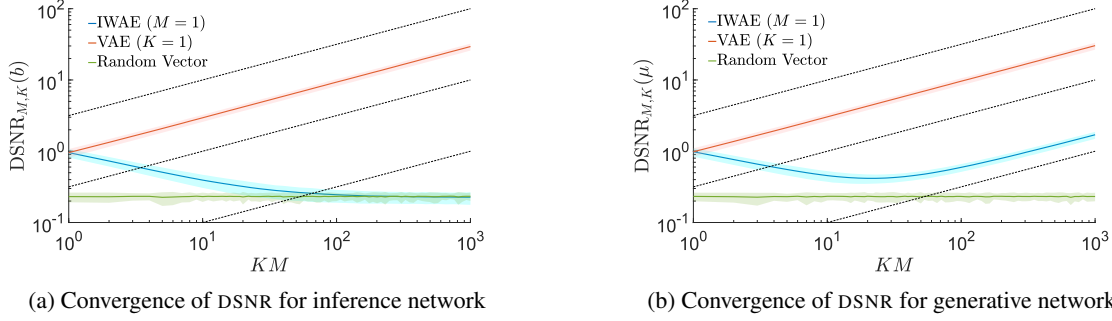


Figure 3: Convergence of the directional SNR of gradients estimates with increasing  $M$  and  $K$ . The solid lines show the estimated DSNR and the shaded regions the interquartile range of the individual ratios. Also shown for reference is the DSNR for a randomly generated vector where each component is drawn from a unit Gaussian.

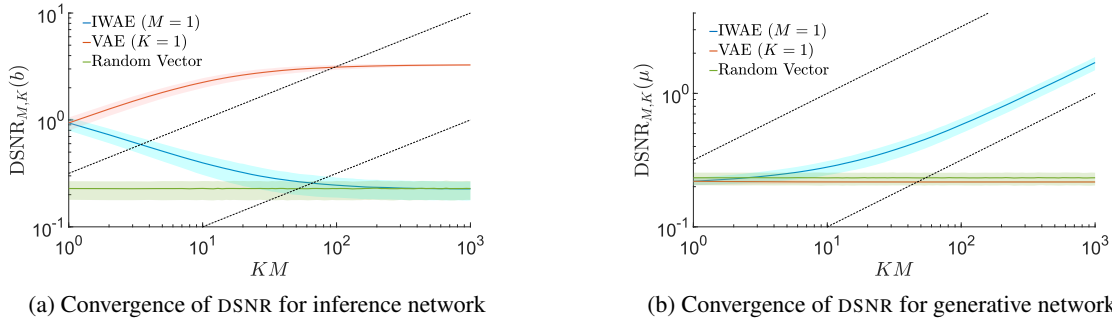


Figure 4: Convergence of the DSNR when the target gradient is taken as  $u = \mathbb{E}[\Delta_{1,1000}]$ . Conventions as per Figure 3.

previous approaches coming from the fact that it takes both  $M > 1$  and  $K > 1$ . The motivation for this is that because our inference network SNR increases as  $O(\sqrt{M/K})$ , we should be able to mitigate the issues increasing  $K$  has on the SNR by also increasing  $M$ . For fairness, we will keep our overall budget  $T = MK$  fixed, but we will show that given this budget, the optimal value for  $M$  is often not 1. In practice, we expect that it will often be beneficial to increase the mini-batch size  $N$  rather than  $M$  for MIWAE; as we showed in Section 3.2 this has the same effect on the SNR. Nonetheless, MIWAE forms an interesting reference method for testing our theoretical results and, as we will show, it can offer improvements over IWAE for a given  $N$ .

Our second algorithm, CIWAE uses a convex combination of the IWAE and VAE bounds, namely

$$\text{ELBO}_{\text{CIWAE}} = \beta \text{ELBO}_{\text{VAE}} + (1 - \beta) \text{ELBO}_{\text{IWAE}} \quad (13)$$

where  $\beta \in [0, 1]$  is a combination parameter. It is trivial to see that  $\text{ELBO}_{\text{CIWAE}}$  is a lower bound on the log marginal that is tighter than the VAE bound but looser than the IWAE bound. We then employ the following estimator

$$\Delta_{K,\beta}^{\text{C}} = \nabla_{\theta,\phi} \left( \beta \frac{1}{K} \sum_{k=1}^K \log w_k + (1 - \beta) \log \left( \frac{1}{K} \sum_{k=1}^K w_k \right) \right) \quad (14)$$

where we use the same  $w_k$  for both terms. The motivation for CIWAE is that, if we set  $\beta$  to a relatively small value, the objective will behave mostly like IWAE, except when

the expected IWAE gradient becomes very small. When this happens, the VAE component should “take-over” and alleviate SNR issues: the asymptotic SNR of  $\Delta_{K,\beta}^{\text{C}}$  for  $\phi$  is  $O(\sqrt{MK})$  because the VAE component has non-zero expectation in the limit  $K \rightarrow \infty$ .

Our results suggest that what is good for the generative network, in terms of setting  $K$ , is often detrimental for the inference network. It is therefore natural to question whether it is sensible to always use the same target for both the inference and generative networks. Motivated by this, our third method, PIWAE, uses the IWAE target when training the generative network, but the MIWAE target for training the inference network. We thus have

$$\Delta_{K,\beta}^{\text{C}}(\theta) = \nabla_{\theta} \log \frac{1}{K} \sum_{k=1}^K w_k \quad (15a)$$

$$\Delta_{M,K,\beta}^{\text{C}}(\phi) = \frac{1}{M} \sum_{m=1}^M \nabla_{\phi} \log \frac{1}{L} \sum_{\ell=1}^L w_{m,\ell} \quad (15b)$$

where we will generally set  $K = ML$  so that the same weights can be used for both gradients.

## 5.1 Experiments

We now use our new estimators to train deep generative models for the MNIST digits dataset (LeCun et al., 1998). For this, we duplicated the architecture and training schedule outlined in Burda et al. (2016). In particular, all networks were trained and evaluated using their stochastic binarization. For all methods we set a budget of  $T = 64$  weights in the target estimate for each datapoint in the minibatch.

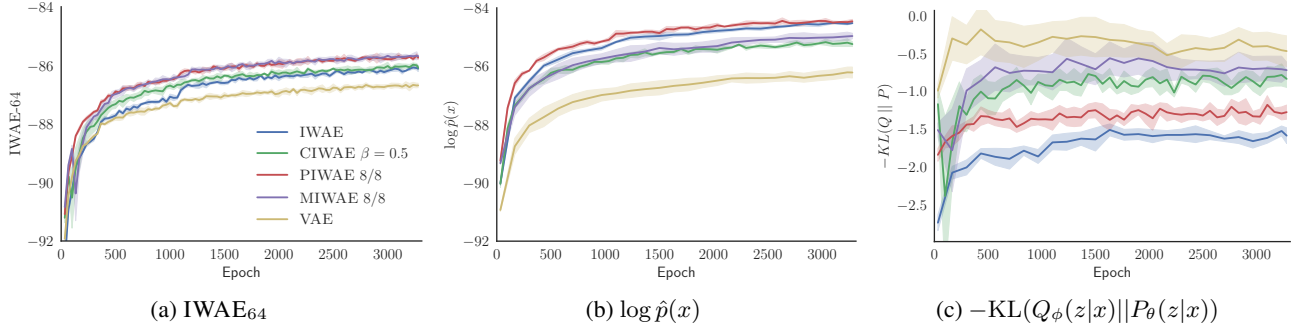


Figure 5: Convergence of evaluation metrics on the test set with increased training time. All lines show mean  $\pm$  standard deviation over 4 runs with different random initializations. Larger values are preferable for each plot.

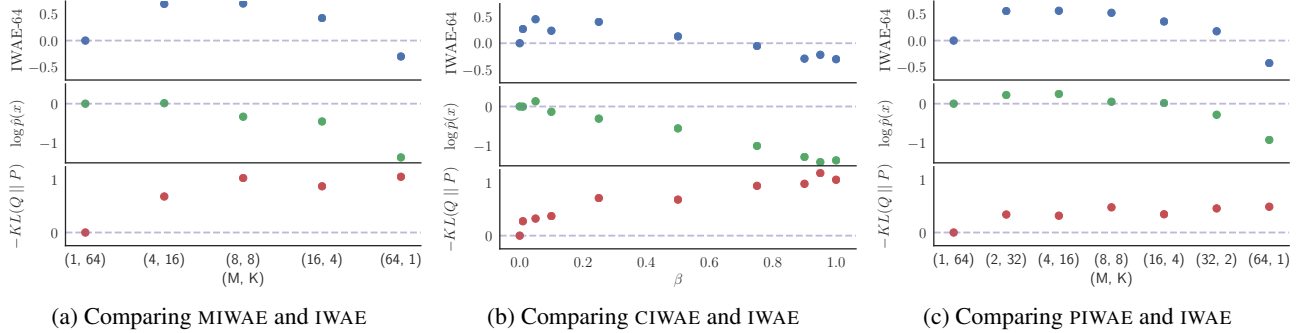


Figure 6: Test set performance of MIWAE, CIWAE, and PIWAE relative to IWAE in terms of the IWAE-64 (top),  $\log \hat{p}(x)$  (middle), and  $-\text{KL}(Q_\phi(z|x)||P_\theta(z|x))$  (bottom) metrics. All dots are the difference in the metric to that of IWAE. Dotted line is the IWAE baseline. Note that in all cases, the far left of the plot correspond to settings equivalent to the IWAE.

Table 1: Mean final test set performance  $\pm$  standard deviation over 4 runs. Numbers in brackets in indicate  $(M, K)$ . The best result is shown in red, while bold results are not statistically significant to best result at the 5% level of a Welch’s t-test.

Metric	IWAE	PIWAE (4, 16)	PIWAE (8, 8)	MIWAE (4, 16)	MIWAE (8, 8)	CIWAE $\beta = 0.05$	CIWAE $\beta = 0.5$	VAE
IWAE-64	-86.11 $\pm$ 0.10	<b>-85.68 <math>\pm</math> 0.06</b>	<b>-85.74 <math>\pm</math> 0.07</b>	<b>-85.60 <math>\pm</math> 0.07</b>	<b>-85.69 <math>\pm</math> 0.04</b>	-85.91 $\pm$ 0.11	-86.08 $\pm$ 0.08	-86.69 $\pm$ 0.08
$\log \hat{p}(x)$	<b>-84.52 <math>\pm</math> 0.02</b>	<b>-84.40 <math>\pm</math> 0.17</b>	<b>-84.46 <math>\pm</math> 0.06</b>	<b>-84.56 <math>\pm</math> 0.05</b>	-84.97 $\pm$ 0.10	<b>-84.57 <math>\pm</math> 0.09</b>	-85.24 $\pm$ 0.08	-86.21 $\pm$ 0.19
$-\text{KL}(Q  P)$	-1.59 $\pm$ 0.10	-1.27 $\pm$ 0.18	-1.28 $\pm$ 0.09	<b>-1.04 <math>\pm</math> 0.08</b>	<b>-0.72 <math>\pm</math> 0.11</b>	-1.34 $\pm$ 0.14	<b>-0.84 <math>\pm</math> 0.11</b>	<b>-0.47 <math>\pm</math> 0.20</b>

To assess different aspects of the training performance, we consider three different metrics:  $\text{ELBO}_{\text{IWAE}}$  with  $K = 64$ ,  $\text{ELBO}_{\text{IWAE}}$  with  $K = 5000$ , and the latter of these minus the former. All reported metrics are evaluated on the test data.

The motivation for the  $\text{ELBO}_{\text{IWAE}}$  with  $K = 64$  metric, denoted as IWAE-64, is that this is the target used for training the IWAE and so if another method does better on this metric than the IWAE, this is a clear indicator that SNR issues of the IWAE estimator have degraded its performance. In fact, this would demonstrate that, from a practical perspective, using the IWAE estimator is sub-optimal, even if our explicit aim is to optimize the IWAE bound. The second metric,  $\text{ELBO}_{\text{IWAE}}$  with  $K = 5000$ , denoted  $\log \hat{p}(x)$ , is used as a surrogate for estimating the log marginal likelihood and thus provides an indicator for fidelity of the learned generative model. The third metric is an estimator for the divergence implicitly targeted by the IWAE. Namely, as shown by Le et al. (2018), the  $\text{ELBO}_{\text{IWAE}}$  can be interpreted as

$$\text{ELBO}_{\text{IWAE}} = \log p_\theta(x) - \text{KL}(Q_\phi(z|x)||P_\theta(z|x)) \quad (16)$$

$$\text{where } Q_\phi(z|x) := \prod_{k=1}^K q_\phi(z_k|x), \text{ and} \quad (17)$$

$$P_\theta(z|x) := \frac{1}{K} \sum_{k=1}^K \frac{\prod_{\ell=1}^K q_\phi(z_\ell|x)}{q_\phi(z_k|x)} p_\theta(z_k|x). \quad (18)$$

Thus we can estimate  $\text{KL}(Q_\phi(z|x)||P_\theta(z|x))$  using  $\log \hat{p}(x) - \text{IWAE-64}$ , to provide a metric for divergence between the inference network and the proposal network. We use this instead of  $\text{KL}(q_\phi(z|x)||p_\theta(z|x))$  because the latter can be deceptive metric for the inference network fidelity. For example, it tends to prefer  $q_\phi(z|x)$  that cover only one of the posterior modes, rather than encompassing all of them. As we showed in Section 3.1, the implied target of the true gradients for the inference network improves as  $K$  increases and so  $\text{KL}(Q_\phi(z|x)||P_\theta(z|x))$  should be a more reliable metric of inference network performance.

Figure 5 shows the convergence of these metrics for each algorithm. Here we have considered the middle value for each of the parameters, namely  $K = M = 8$  for PIWAE and MIWAE, and  $\beta = 0.5$  for CIWAE. We see that PIWAE and MIWAE both comfortably outperformed, and CIWAE

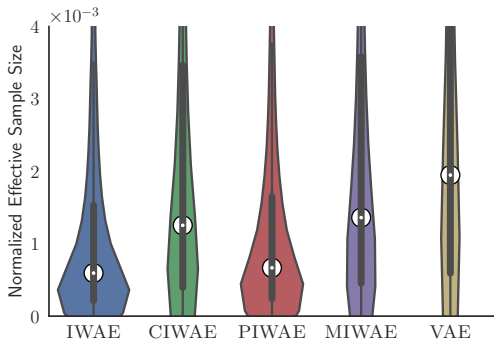


Figure 7: Violin plots of ESS estimates for each image of MNIST, normalized by the number of samples drawn. A violin plot uses a kernel density plot on each side – thicker means more MNIST images whose  $q_\phi$  achieves that ESS.

slightly outperformed, IWAE in terms of IWAE-64 metric, despite IWAE being directly trained on this target. In terms of  $\log \hat{p}(x)$ , PIWAE gave the best performance, followed by IWAE. For the KL, we see that the VAE performed best followed by MIWAE, with IWAE performing the worst. We note here that the KL is not an exact measure of the inference network performance as it also depends on the generative model. As such, the apparent superior performance of the VAE may be because it produces a simpler model, as per the observations of Burda et al. (2016), which in turn is easier to learn an inference network for. Critically though, PIWAE improves this metric whilst also improving generative network performance, such that this reasoning no longer applies. Similar behavior is observed for MIWAE and CIWAE for different parameter settings (see Appendix D).

We next considered tuning the parameters for each of our algorithms as shown in Figure 6, for which we look at the final metric values after training. Table 1 further summarizes the performance for certain selected parameter settings. For MIWAE we see that as we increase  $M$ , the  $\log \hat{p}(x)$  metric gets worse, while the KL gets better. The IWAE-64 metric initially increases with  $M$ , before reducing again from  $M = 16$  to  $M = 64$ , suggesting that intermediate values for  $M$  (i.e.  $M \neq 1$ ,  $K \neq 1$ ) give a better trade-off. For PIWAE, similar behavior to MIWAE is seen for the IWAE-64 and KL metrics. However, unlike for MIWAE, we see that  $\log \hat{p}(x)$  initially increases with  $M$ , such that PIWAE provides uniform improvement over IWAE for the  $M = 2, 4, 8$ , and 16 cases. CIWAE exhibits similar behavior in increasing  $\beta$  as increasing  $M$  for MIWAE, but there appears to be a larger degree of noise in the evaluations, while the optimal value of  $\beta$ , though non-zero, seems to be closer to IWAE than for the other algorithms.

As an additional measure of the performance of the inference network that is distinct to any of the training targets, we also considered the effective sample size (ESS) (Owen, 2013) for the fully trained networks, defined as

$$\text{ESS} = (\sum_{k=1}^K w_k)^2 / \sum_{k=1}^K w_k^2. \quad (19)$$

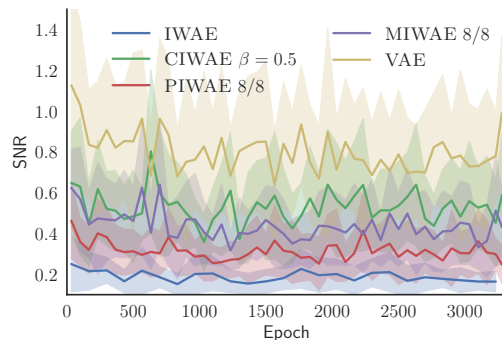


Figure 8: SNR of inference network weights during training. All lines are mean  $\pm$  standard deviation over 20 randomly chosen weights per layer.

The ESS is a measure of how many unweighted samples would be equivalent to the weighted sample set. A low ESS indicates that the inference network is struggling to perform effective inference for the generative network. The results, given in Figure 7, show that the ESSs for CIWAE, MIWAE, and the VAE were all significantly larger than for IWAE and PIWAE, with IWAE giving a particularly poor ESS.

Our final experiment looks at the SNR values for the inference networks during training. Here we took a number of different neural network gradient weights at different layers of the network and calculated empirical estimates for their SNRs at various points during the training. We then averaged these estimates over the different network weights, the results of which are given in Figure 8. This clearly shows the low SNR exhibited by the IWAE inference network, suggesting that our results from the simple Gaussian experiments carry over to the more complex neural network domain.

## 6 Conclusions

We have provided theoretical and empirical evidence that algorithmic approaches to increasing the tightness of the ELBO independently to the expressiveness of the inference network can be detrimental to learning by reducing the signal-to-noise ratio of the inference network gradients. Experiments on a simple latent variable model confirmed our theoretical findings. We then exploited these insights to introduce three estimators, PIWAE, MIWAE, and CIWAE and showed that each can deliver improvements over IWAE, even when the metric used for this assessment is the IWAE target itself. In particular, each was able to deliver improvement in the training of the inference network, without any reduction in the quality of the learned generative network.

Whereas MIWAE and CIWAE mostly allow for balancing the requirements of the inference and generative networks, PIWAE appears to be able to offer simultaneous improvements to both, with the improved training of the inference network having a knock-on effect on the generative network. Key to achieving this is, is its use of separate targets for the two networks, opening up interesting avenues for future work.



## Acknowledgments

TR and YWT are supported in part by the European Research Council under the European Union’s Seventh Framework Programme (FP7/2007–2013) / ERC grant agreement no. 617071. TAL is supported by a Google studentship, project code DF6700. MI is supported by the UK EPSRC CDT in Autonomous Intelligent Machines and Systems. CJM is funded by a DeepMind Scholarship. FW is supported under DARPA PPAML through the U.S. AFRL under Cooperative Agreement FA8750-14-2-0006, Sub Award number 61160290-111668.

## References

- Bamler, R., Zhang, C., Opper, M., and Mandt, S. Perturbative black box variational inference. *arXiv preprint arXiv:1709.07433*, 2017.
- Bouillard, H. and Kamp, Y. Auto-association by multilayer perceptrons and singular value decomposition. *Biological cybernetics*, 59(4-5):291–294, 1988.
- Burda, Y., Grosse, R., and Salakhutdinov, R. Importance weighted autoencoders. In *ICLR*, 2016.
- Chen, X., Kingma, D. P., Salimans, T., Duan, Y., Dhariwal, P., Schulman, J., Sutskever, I., and Abbeel, P. Variational lossy autoencoder. In *ICLR*, 2017.
- Cremer, C., Morris, Q., and Duvenaud, D. Reinterpreting importance-weighted autoencoders. *arXiv preprint arXiv:1704.02916*, 2017.
- Fort, G., Gobet, E., and Moulines, E. Mcmc design-based non-parametric regression for rare event. application to nested risk computations. *Monte Carlo Methods and Applications*, 23(1):21–42, 2017.
- Gregor, K., Besse, F., Rezende, D. J., Danihelka, I., and Wierstra, D. Towards conceptual compression. In *NIPS*, 2016.
- Hesterberg, T. C. *Advances in importance sampling*. PhD thesis, Stanford University, 1988.
- Hinton, G. E. and Zemel, R. S. Autoencoders, minimum description length and helmholtz free energy. In *NIPS*, 1994.
- Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.
- Kingma, D. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Kingma, D. P. and Welling, M. Auto-encoding variational Bayes. In *ICLR*, 2014.
- Kingma, D. P., Salimans, T., and Welling, M. Improving variational inference with inverse autoregressive flow. *arXiv preprint arXiv:1606.04934*, 2016.
- Le, T. A., Baydin, A. G., and Wood, F. Inference compilation and universal probabilistic programming. In *AISTATS*, 2017.
- Le, T. A., Igl, M., Rainforth, T., Jin, T., and Wood, F. Auto-encoding sequential Monte Carlo. In *ICLR*, 2018.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- Li, Y. and Turner, R. E. Rényi divergence variational inference. In *NIPS*, 2016.
- Maaløe, L., Sønderby, C. K., Sønderby, S. K., and Winther, O. Auxiliary deep generative models. *arXiv preprint arXiv:1602.05473*, 2016.
- Maddison, C. J., Lawson, D., Tucker, G., Heess, N., Norouzi, M., Mnih, A., Doucet, A., and Teh, Y. W. Filtering variational objectives. *arXiv preprint arXiv:1705.09279*, 2017.
- Naesseth, C. A., Linderman, S. W., Ranganath, R., and Blei, D. M. Variational sequential Monte Carlo. 2018.
- Owen, A. B. *Monte Carlo theory, methods and examples*. 2013.
- Paige, B. and Wood, F. Inference networks for sequential Monte Carlo in graphical models. In *ICML*, 2016.
- Rainforth, T. *Automating Inference, Learning, and Design using Probabilistic Programming*. PhD thesis, 2017.
- Rainforth, T., Cornish, R., Yang, H., Warrington, A., and Wood, F. On nesting Monte Carlo estimators. In *ICML*, 2018.
- Ranganath, R., Gerrish, S., and Blei, D. Black box variational inference. In *AISTATS*, 2014.
- Ranganath, R., Tran, D., and Blei, D. Hierarchical variational models. In *ICML*, 2016.
- Rezende, D. and Mohamed, S. Variational inference with normalizing flows. In *International Conference on Machine Learning*, pp. 1530–1538, 2015.
- Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 2014.
- Roberts, J. W. and Tedrake, R. Signal-to-noise ratio analysis of policy gradient algorithms. In *NIPS*, 2009.
- Salimans, T., Kingma, D., and Welling, M. Markov chain Monte Carlo and variational inference: Bridging the gap. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pp. 1218–1226, 2015.
- Tran, D., Ranganath, R., and Blei, D. M. The variational Gaussian process. *arXiv preprint arXiv:1511.06499*, 2015.
- Turner, R. E. and Sahani, M. Two problems with variational expectation maximisation for time-series models. *Bayesian Time series models*, pp. 115–138, 2011.