Table 1: Lowerbounds and NLL for various continuous latent variable models and training algorithms along with the corresponding VAE estimates. We use 200 latent Gaussian variables. †From Salimans et al. [6]. §From Rezende and Mohamed [5]. ‡From Burda et al. [3].

Model	$\leq$ -log p(x)	$\approx$ -log p(x)
VAE	94.48	89.31
VAE (w/ refinement)	90.57	88.53
$\text{GDIR}_{50,20}$	90.60	88.54
VAE†	94.18	88.95
HVI_1†	91.70	88.08
HVI_8†	88.30	85.51
VAE §	89.9	
DLGM+NF_{80} §	**85.1**	
VAE‡		86.35
IWAE $(K = 50)$‡		**84.78**

1 Supplementary material

1.1 Continuous Variables

With variational autoencoders (VAE), the back-propagated gradient of the lowerbound with respect to the approximate posterior is composed of individual gradients for each factor, μ_i that can be applied simultaneously. Applying the gradient directly to the variational parameters, $\boldsymbol{\mu}$, without back-propagating to the recognition network parameters, ψ, yields a simple iterative refinement operator:

$$\boldsymbol{\mu}_{t+1} = g(\boldsymbol{\mu}_t, \mathbf{x}, \gamma) = \boldsymbol{\mu}_t + \gamma \nabla_{\boldsymbol{\mu}} \mathcal{L}_1(\boldsymbol{\mu}, \mathbf{x}, \boldsymbol{\epsilon}), \tag{1}$$

where γ is the inference rate hyperparameter and $\boldsymbol{\epsilon}$ is auxiliary noise used in the re-parameterization.

This gradient-descent iterative refinement (GDIR) is very straightforward with continuous latent variables as with VAE. However, GDIR with discrete units suffers the same shortcomings as when passing the gradients directly, so a better transition operator is needed (AIR).

In the limit of $T = 0$, we do not arrive at VAE, as the gradients are never passed through the approximate posterior during learning. However, as the complete computational graph involves a series of differentiable variables, $\boldsymbol{\mu}_t$, in addition to auxiliary noise, it is possible to pass gradients through GDIR to the recognition network parameters, ψ, during learning, though we do not here.

For continuous latent variables, we used the same network structure as in [4, 6]. Results for GDIR are presented in Table 1 for the MNIST dataset, and included for comparison are methods for learning non-factorial latent distributions for Gaussian variables and the corresponding result for VAE, the baseline.

Though GDIR can improve the posterior in VAE, our results show that VAE is at an upper-bound for learning with a factorized posterior on the MNIST dataset. Further improvements on this dataset must be made by using a non-factorized posterior (re-weighting or sequential Monte Carlo with importance weighting). GDIR may still also provide improvement for training models with other datasets, and we leave this for future work.

2 Refinement of the lowerbound and effective sample size

Iterative refinement via adaptive inference refinement (AIR) improves the variational lowerbound and effective sample size (ESS) of the approximate posterior. To show this, we trained models with one, two, and three hidden layers with 200 binary units trained using AIR with 20 inference steps on the MNIST dataset for 500 epochs. Taking the initial approximate posterior from each model, we refined the posterior up to 50 steps (Figure 1), evaluating the lowerbound and ESS using 100 posterior samples. Refinement improves the posterior from models trained on AIR well beyond the number of steps used in training.

1
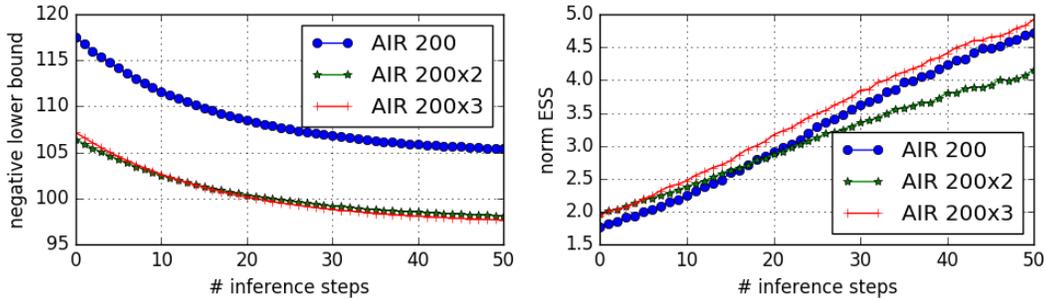
Figure 1: The variational lowerbound (left) and normalized effective sample size (ESS, right) the test set as the posterior is refined from the initial posterior provided by the recognition network. Models were trained with AIR with 20 refinement steps and one (AIR 200), two (AIR 200x2), and three (AIR 200x3) hidden layers. Refinement shows clear improves of both the variational lowerbound and effective sample size.
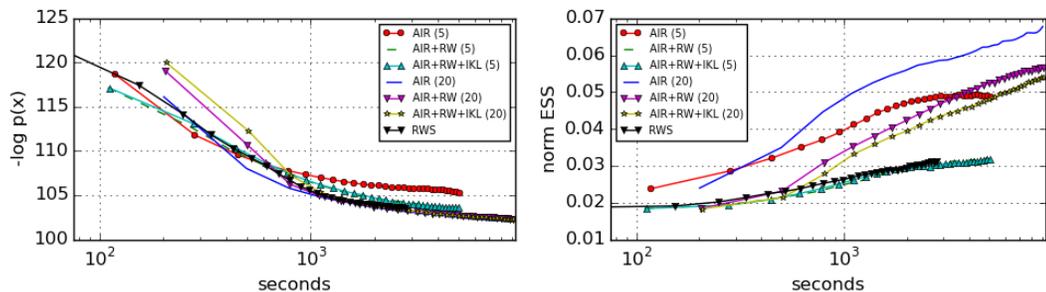
Figure 2: The log-likelihood (left) and normalized effective sample size (right) with epochs in log-scale on the training set for AIR with 5 and 20 refinement steps (vanilla AIR), reweighted AIR with 5 and 20 refinement steps, reweighted AIR with inclusive KL objective and 5 or 20 refinement steps, and reweighted wake-sleep (RWS). Despite longer wall-clock time per epoch, AIR converges to lower log-likelihoods and effective sample size (ESS) than RWS.

3   Updates and wall-clock times

Adaptive iterative refinement (AIR) and reweighted wake-sleep [RWS, 1] have competing convergence wall-clock times, while AIR outperforms on updates (Figures 2 and 3). AIR converges to a higher lowerbound and with far fewer updates than RWS, though RWS converges sooner to a similar value as AIR does later in training time. AIR outperforms RWS in ESS in both wall-clock time and updates. For a more accurate comparison, RWS may need to be trained at wall-clock times equal to that afforded to AIR. However, these results support the conclusion that AIR converges to similar values as RWS in less updates but similar wall-clock times.

4   Bidirectional Helmholtz machines and AIR

As an alternative to the variational lowerbound, a lowerbound can be formulated from the geometric mean of the joint generative and approximate posterior models:

$$p^\star(\mathbf{x}, \mathbf{h}) = \frac{1}{Z}\sqrt{p(\mathbf{x}, \mathbf{h})q(\mathbf{x}, \mathbf{h})}. \tag{2}$$

In this procedure, known as bidirectional Helmholtz machines [2], the lowerbound, which minimizes the Bhattacharyya distance ($D_B(p, q) = -\log \sum_y \sqrt{p(y)q(y)}$), yields estimates of the likelihood, $p^\star(\mathbf{x})$, with importance weights,

$$w^{(k)} = \sqrt{\frac{p(\mathbf{x}, \mathbf{h}^{(k)})}{q(\mathbf{h}^{(k)}|\mathbf{x})}}. \tag{3}$$

2
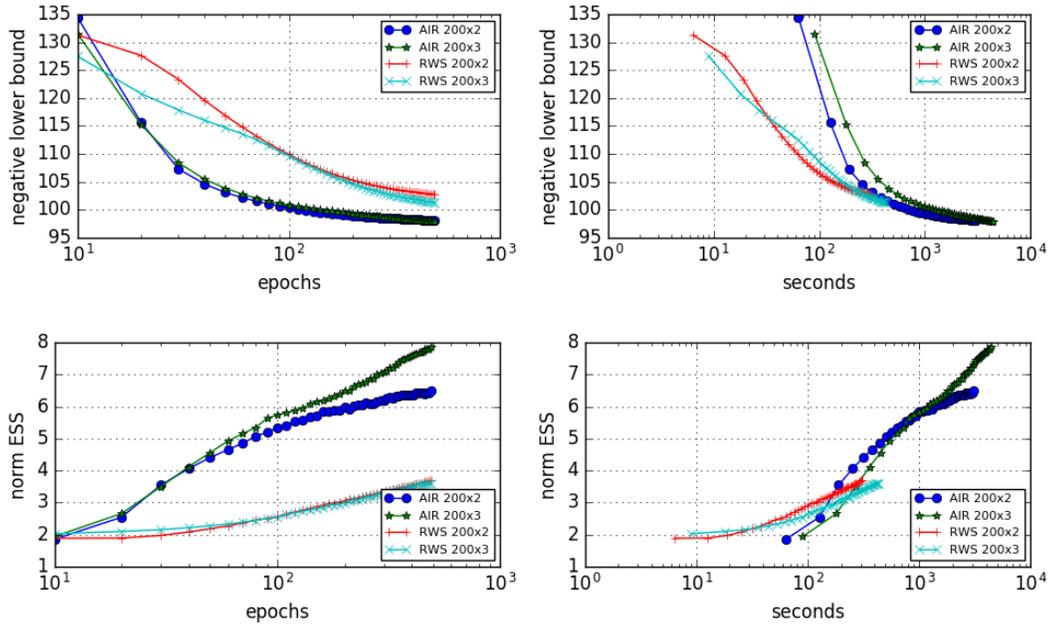
Figure 3: Negative lowerbound and effective samples size (ESS) across updates (epochs) and wall-clock time (seconds) for two and three layer sigmoid belief networks trained with adaptive iterative refinement (AIR) and reweighted wake-sleep (RWS). AIR was trained with 20 refinement steps with a damping rate of $\gamma = 0.9$. Each model was trained for 500 epochs and evaluated on the training dataset using 100 posterior samples. AIR takes less updates to reach equivalent variational lowerbound and ESS than RWS. While RWS can reach a higher lowerbound at earlier wall-clock times, AIR and RWS appear to converge to the same value, and AIR reaches much higher ESS.

Similar to with the variational lowerbound, we can refine the approximate posterior to maximize this lowerbound by simply replacing the weights in Equation 3.

We performed similar experiments to those as the experiments on wall-clock times above, using only a three layer SBN trained for 500 epochs with the equivalent AIR and BiHM procedures using the bidirectional lowerbound importance weights. We evaluated these models using 10000 posterior samples on the test dataset and evaluated BiHM with (BiHM+) and without refinement.

Our results show similar negative log likelihoods for AIR (92.40 nats), BiHM (93.30 nats), and BiHM+ (92.90 nats), though AIR slightly outperforms BiHM+, and BiHM+ slightly outperforms BiHM. Further optimization is necessary for a better comparison to our experiments with the variational lowerbound. However these observations are consistent with those from our original experiments: AIR can be used to improve the posterior both in training and when evaluating models, regardless of how they were trained. Furthermore, AIR is compatible with optimizations based on alternative lowerbounds, broadening the scope in which AIR is applicable.

References

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

[2] Jorg Bornschein, Samira Shabanian, Asja Fischer, and Yoshua Bengio. Bidirectional helmholtz machines. *arXiv preprint arXiv:1506.03877*, 2015.

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

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

[5] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

3

[6] Tim Salimans, Diederik Kingma, and Max Welling. Markov chain monte carlo and variational inference: Bridging the gap. In David Blei and Francis Bach, editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 1218–1226. JMLR Workshop and Conference Proceedings, 2015. URL `http://jmlr.org/proceedings/papers/v37/salimans15.pdf`.

4