Inference Suboptimality in Variational Autoencoders

Chris Cremer, Xuechen Li, David Duvenaud

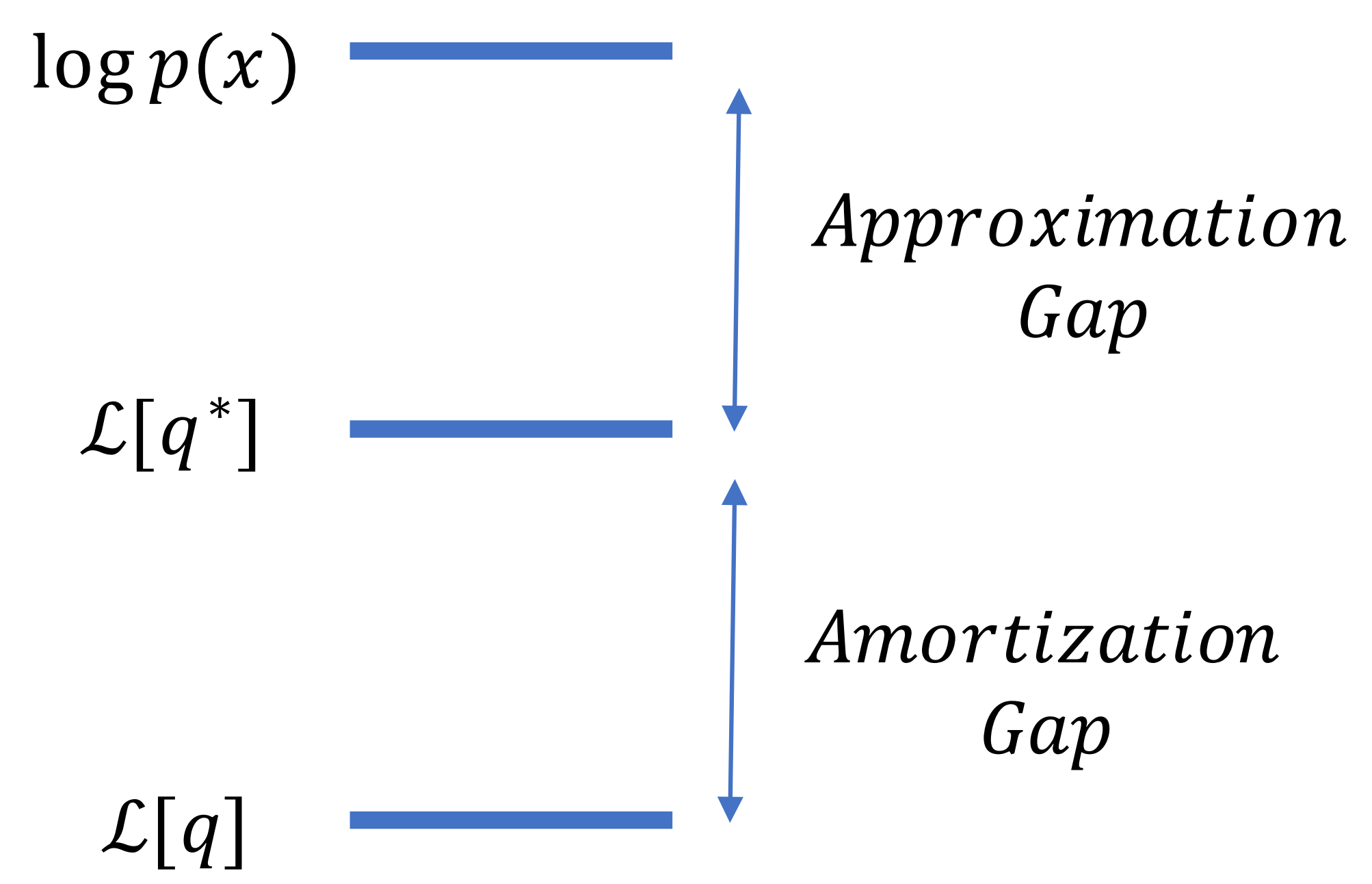
Department of Computer Science, University of Toronto

Main Idea

When inference in VAEs is suboptimal, what is the main cause? We break down the gap between suboptimal and optimal inference into two components: amortization and approximation.

- The amortization gap comes from the limited capacity of the recognition network to generalize inference over all datapoints.
- The approximation gap arises due to the approximate distribution's inability to fit to the true posterior.

This analysis also demonstrates the impact that the choice of the approximate posterior has on the true posterior.



$\log p(x)$

Approximation Gap

$\mathcal{L}[q^*]$

Amortization Gap

$\mathcal{L}[q]$

Approximate Posteriors

A typical choice for the approximate posterior distribution is a fully factorized Gaussian (FFG) distribution. We also examine the behaviour of a more flexible distribution (Flow). Our choice of expressive distribution is a combination of Real NVP [3] and auxiliary variables [5, 4]. The resulting lower bound that is optimized is:

$$\mathbb{E}_{z_0,v_0\sim q(z,v|x)}\left[\log\left(\frac{p(x,z_T)r(v_T|x,z_T)}{q(z_0|x,v_0)q(v_0|x)\prod_{t=1}^{T}\left|\det\frac{\partial z_t v_t}{\partial z_{t-1}v_{t-1}}\right|^{-1}}\right)\right].$$

Inference Gap Breakdown

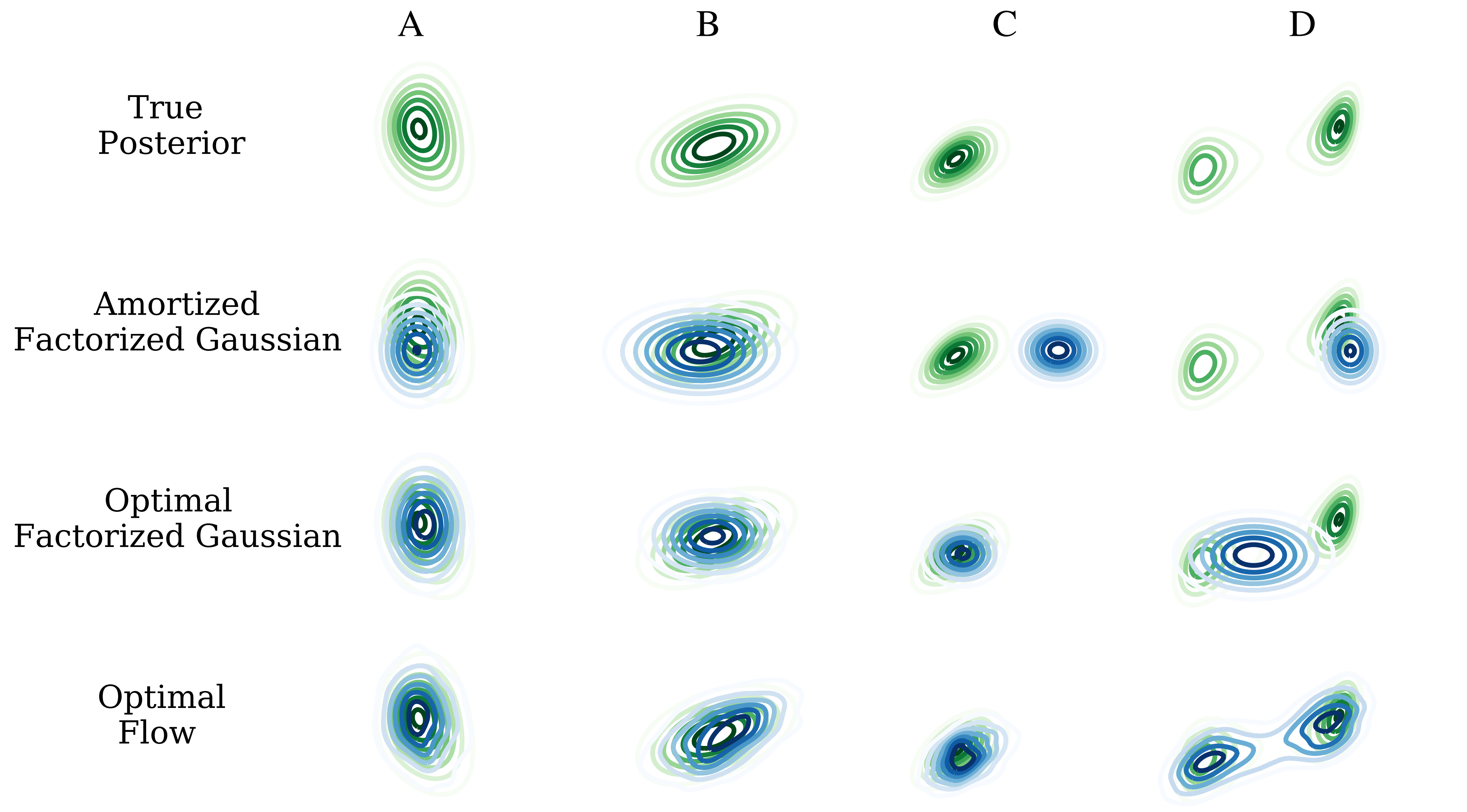
The inference gap $\mathcal{G}$ is the difference between the marginal log-likelihood $\log p(x)$ and its lower bound $\mathcal{L}[q]$. Given the distribution in the variational family that maximizes the lower bound, $q^*(z|x) = \arg\max_{q\in\mathcal{Q}} \mathcal{L}[q]$, the inference gap can be decomposed as the sum of the approximation and amortization gaps:

$$\mathcal{G} = \log p(x) - \mathcal{L}[q] = \underbrace{\log p(x) - \mathcal{L}[q^*]}_{\text{Approximation}} + \underbrace{\mathcal{L}[q^*] - \mathcal{L}[q]}_{\text{Amortization}}.$$

For VAEs, we can translate the gaps to KL divergences:

$$\mathcal{G}_{\text{VAE}} = \underbrace{\text{KL}\big(q^*(z|x)||p(z|x)\big)}_{\text{Approximation}} + \underbrace{\text{KL}\big(q(z|x)||p(z|x)\big) - \text{KL}\big(q^*(z|x)||p(z|x)\big)}_{\text{Amortization}}.$$

The figure below is a visualization of the latent space of a 2D VAE trained on MNIST. Examples A through D were selected to highlight particular instances of inference suboptimality. C highlights the amortization gap since the amortized FFG (blue) shares little support with the true posterior (green). The true posterior of D is bimodal which demonstrates the ability of the flexible approximation to fit to complex distributions, in contrast to the simple FFG approximation, highlighting the approximation gap.



Amortization vs Approximation

How much of the inference gap is due to amortization compared to the gap caused by the posterior approximation? Table 1 are results from a model with the same architectures as [2] trained on MNIST and Fashion MNIST.

- On MNIST, both the amortization and approximation gaps contribute roughly equally to the inference gap.
- On Fashion MNIST, the majority of inference gap is due to amortization. This suggests that the encoder could benefit from increased capacity. To confirm this, we increase the size of the encoder (*Large* in Table 1) and we see a decrease in the amortization gap.

Approximation Influences True

To what extent does the choice of approximation affect the true posterior? We can quantitatively determine how close the posterior is to a FFG distribution by comparing the Optimal FFG bound $\mathcal{L}_{\text{VAE}}[q^*_{FFG}]$ and the Optimal Flow bound $\mathcal{L}_{\text{VAE}}[q^*_{Flow}]$: their difference measures the improvement gained from the flexibility of a Flow distribution over a FFG. From Table 1, we observe that:

- for models trained with q_{FFG}, the difference between $\mathcal{L}_{\text{VAE}}[q^*_{FFG}]$ and $\mathcal{L}_{\text{VAE}}[q^*_{Flow}]$ is at most 2.2 nats
- for models trained with q_{Flow}, the difference between $\mathcal{L}_{\text{VAE}}[q^*_{FFG}]$ and $\mathcal{L}_{\text{VAE}}[q^*_{Flow}]$ is more than 10 nats

This suggests that training with a flexible approximation results in complex posteriors whereas training with a FFG approximation results in posteriors that are closer to a FFG.

Annealing Entropy

Typical warm-up [1, 6] is known to help prevent the latent variables from degrading to the prior [2, 6]. We employ a similar technique during training, where the entropy of q is annealed:

$$\mathbb{E}_{z\sim q(z|x)}\left[\log p(x,z) - \lambda \log q(z|x)\right],$$

where λ is annealed from 0 to 1 over training. We find that this is very important for allowing the true posterior to be more complex. *No Anneal* of Table 1 refers to models trained without the entropy annealing. We see that the difference between $\mathcal{L}_{\text{VAE}}[q^*_{FFG}]$ and $\mathcal{L}_{\text{VAE}}[q^*_{Flow}]$ is significantly smaller without warmup than with warmup. This suggests that, in addition to preventing the latent variable from degrading to the prior, warmup allows the true posterior to better utilize the flexibility of the expressive approximation, resulting in a better trained model.

Experimental Results

Dataset	MNIST						Fashion MNIST					
Encoder Capacity	Regular		Large		No Anneal		Regular		Large		No Anneal	
Variational Family	q_{FFG}	q_{Flow}	q_{FFG}	q_{Flow}	q_{FFG}	q_{Flow}	q_{FFG}	q_{Flow}	q_{FFG}	q_{Flow}	q_{FFG}	q_{Flow}
$\log \hat{p}(x)$	-89.85	-88.82	-89.09	-88.59	-89.84	-89.41	-97.78	-97.35	-94.55	-96.16	-100.56	-100.33
$\mathcal{L}_{\text{VAE}}[q^*_{Flow}]$	-90.83	-90.40	-90.05	-90.26	-90.89	-90.78	-98.03	-97.74	-95.93	-96.87	-100.74	-100.61
$\mathcal{L}_{\text{VAE}}[q^*_{FFG}]$	-91.19	-102.88	-90.34	-103.53	-91.02	-92.34	-99.03	-130.90	-98.09	-129.24	-101.26	-102.80
$\mathcal{L}_{\text{VAE}}[q]$	-92.76	-91.42	-91.12	-91.25	-94.31	-94.29	-103.20	-102.19	-101.28	-100.60	-103.63	-104.05
Approximation	1.34	1.58	1.25	1.67	1.18	1.37	1.25	0.39	3.54	0.71	0.70	0.28
Amortization	1.57	1.02	0.78	0.99	3.29	3.51	4.17	4.45	3.19	3.73	2.37	3.44
Inference Gap	2.91	2.60	2.03	2.66	4.47	4.88	5.42	4.84	6.73	4.44	3.07	3.72

Table 1: Lower bounds and inference gaps. *Variational Family* refers to the approximate distribution used to train the model. *No Anneal* columns use *Regular* encoder capacity. All numbers are in nats.

References

[1] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio. Generating Sentences from a Continuous Space. *ArXiv e-prints*, November 2015.

[2] Y. Burda, R. Grosse, and R. Salakhutdinov. Importance weighted autoencoders. *In ICLR*, 2016.

[3] L. Dinh, J. Sohl-Dickstein, and S. Bengio. Density estimation using Real NVP. *ICLR*, 2017.

[4] L. Maaløe, CK. Sønderby, SK. Sønderby, and O. Winther. Auxiliary Deep Generative Models. *ICML*, 2016.

[5] R. Ranganath, D. Tran, and D. M. Blei. Hierarchical Variational Models. *ICML*, 2016.

[6] C.K. Sønderby, T. Raiko, L. Maaløe, S. Kaae Sønderby, and O. Winther. Ladder Variational Autoencoders. *CONF*, 2016.