Refit your Encoder when New Data Comes by

Pierre-Alexandre Mattei
Department of Computer Science
IT University of Copenhagen
pima@itu.dk

Jes Frellsen
Department of Computer Science
IT University of Copenhagen
jefr@itu.dk

Abstract

Deep latent variable models (DLVMs) are flexible generative models whose likelihood is often intractable. Such models can be trained thanks to an encoder (or inference network) that plays the role of an approximation of the posterior distribution of the latent variables. Conveniently, this encoder can also be used to estimate the likelihood, by using it as a proposal in importance sampling. When assessing the likelihood of out-of-sample data, it is common practice not to refit the encoder to the new data. We point out the drawbacks of this practice and advocate for refitting the encoder as a simple refinement the popular importance sampling estimate of the likelihood.

1 Deep latent variable models and their likelihood

Deep latent variable models (DLVMs, Rezende et al., 2014; Kingma and Welling, 2014) are generative models that draw their flexibility from deep architectures. DLVMs assume that $(\mathbf{x}_i, \mathbf{z}_i)_{i \leq n} \in (\mathcal{X} \times \mathbb{R}^d)^n$ are i.i.d. random variables driven by the following generative model:

$$\begin{cases} \mathbf{z} \sim p(\mathbf{z}) \\ p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) = \Phi(\mathbf{x}|f_{\boldsymbol{\theta}}(\mathbf{z})). \end{cases} \tag{1}$$

The low-dimensional hidden *codes* $\mathbf{z} \in \mathbb{R}^d$ are passed though a function $f_{\boldsymbol{\theta}} : \mathbb{R}^d \rightarrow H$, called the *decoder* or *generative network*, parametrised by a neural network whose weights are stored in $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. The set $(\Phi(\cdot|\boldsymbol{\eta}))_{\boldsymbol{\eta} \in H}$, called the *observation model*, is a parametric family of densities with respect to a dominating measure (usually the Lebesgue or the counting measure).

The marginal distribution of the data is then a rather difficult quantity to compute because of the integration over the latent space. Consequently, Monte Carlo estimation though importance sampling is often performed to estimate the likelihood. The optimal proposal for importance sampling would be the posterior distribution of the code, which is also likely not to be amenable. However, this complex posterior can be approximated by a tractable conditional distribution $q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}) = \Psi(\mathbf{z}|g_{\boldsymbol{\gamma}}(\mathbf{x}))$, where $(\Psi(\cdot|\boldsymbol{\kappa}))_{\boldsymbol{\kappa} \in K}$ is a family of densities over $\mathbb{R}^d$, and $g_{\boldsymbol{\gamma}} : \mathcal{X} \rightarrow K$ is a neural network called the *encoder*, or *inference network*, whose weights are stored in $\boldsymbol{\gamma} \in \boldsymbol{\Gamma}$. Conveniently, estimates of both $\boldsymbol{\theta}$ and $\boldsymbol{\gamma}$ can be found by maximising the function

$$\mathcal{L}(\boldsymbol{\theta}, \gamma) = \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_i)} \left[\log \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_i, \mathbf{z})}{q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_i)} \right], \tag{2}$$

often called the *evidence lower bound* (ELBO). Indeed, the identity

$$\mathcal{L}(\boldsymbol{\theta}, \gamma) = \sum_{i=1}^{n} \log p_{\boldsymbol{\theta}}(\mathbf{x}_i) - \sum_{i=1}^{n} \mathrm{KL}(q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_i) || p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x}_i)), \tag{3}$$

Third workshop on Bayesian Deep Learning (NeurIPS 2018), Montréal, Canada.

implies that maximising $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\gamma})$ will both maximise a lower bound of the likelihood, and therefore find a suitable $\boldsymbol{\theta}$, and minimise the Kullback-Leibler divergences between the true posteriors and the distributions $q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_1), \ldots, q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_n)$, which will will ensure that $q_{\boldsymbol{\gamma}}$ is a valid approximation of the posterior. Using $q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x})$ as an importance sampling proposal leads to the estimate

$$p_{\boldsymbol{\theta}}(\mathbf{x}) \approx \frac{1}{K} \sum_{k=1}^{K} \frac{p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}_k) p(\mathbf{z}_k)}{q_{\boldsymbol{\gamma}}(\mathbf{z}_k|\mathbf{x})}, \tag{4}$$

where $\mathbf{z}_1, ..., \mathbf{z}_K$ are i.i.d. samples from $q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x})$. As suggested by Burda et al. (2016), this estimate can also be used to define a tighter lower bound of the log-likelihood than the one given in Eq. (2). This leads to the *importance weighted autoencoder* (IWAE) objective function

$$\mathcal{L}_K(\boldsymbol{\theta}, \boldsymbol{\gamma}) = \sum_{i=1}^{n} \mathbb{E}_{\mathbf{z}_1,\ldots,\mathbf{z}_K \sim q_{\boldsymbol{\gamma}}(\mathbf{z}|\mathbf{x}_i)} \left[\log \frac{1}{K} \sum_{k=1}^{K} \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_i|\mathbf{z}_k) p(\mathbf{z}_k)}{q_{\boldsymbol{\gamma}}(\mathbf{z}_k|\mathbf{x}_i)} \right]. \tag{5}$$

While the IWAE objective is trickier to optimise (Tucker et al., 2018; Rainforth et al., 2018), it gives a provably more accurate estimate of the log-likelihood that converges to the exact log-likelihood as $K \to \infty$ (Burda et al., 2016; Nowozin, 2018).

2 Likelihood evaluation for out-of-sample data

Assume that we have trained a DLVM and have access to a couple encoder/decoder $(f_{\boldsymbol{\theta}^*}, g_{\boldsymbol{\gamma}^*})$. Let $\tilde{\mathbf{x}}$ be a new, unseen data point (for example a test sample). An important quantity in generative modelling is the likelihood $p_{\boldsymbol{\theta}^*}(\tilde{\mathbf{x}})$ of this new data point. For example, the test likelihood is commonly used as a performance metric for generative models and in particular for DLVMs, see e.g. Rezende et al. (2014); Salimans et al. (2015); Rezende et al. (2016).

Rezende et al. (2014, Appendix E) suggested to use the trained encoder to estimate $p_{\boldsymbol{\theta}^*}(\tilde{\mathbf{x}})$, leading to

$$p_{\boldsymbol{\theta}^*}(\tilde{\mathbf{x}}) \approx \frac{1}{K} \sum_{k=1}^{K} \frac{p_{\boldsymbol{\theta}^*}(\tilde{\mathbf{x}}|\mathbf{z}_k) p(\mathbf{z}_k)}{q_{\boldsymbol{\gamma}^*}(\mathbf{z}_k|\tilde{\mathbf{x}})}, \tag{6}$$

with $\mathbf{z}_1, ..., \mathbf{z}_K \sim q_{\boldsymbol{\gamma}^*}(\mathbf{z}|\tilde{\mathbf{x}})$. As we argued in Section 1, for any training data point $\mathbf{x}_i$ ($i \in \{1, ..., n\}$), there are good reasons to believe that $q_{\boldsymbol{\gamma}^*}(\mathbf{z}|\mathbf{x}_i)$ is a good proposal to perform importance sampling in order to estimate $p_{\boldsymbol{\theta}^*}(\mathbf{x}_i)$. However, due to potential overfitting of the decoder, $q_{\boldsymbol{\gamma}^*}(\mathbf{z}|\tilde{\mathbf{x}})$ might be a poor proposal when it comes to estimate $p_{\boldsymbol{\theta}^*}(\tilde{\mathbf{x}})$, as exhibited by Wu et al. (2017) and Cremer et al. (2018). Consequently, Wu et al. (2017) and Cremer et al. (2018) suggested to used annealed importance sampling (AIS), and Nowozin (2018) also proposed a refinement of the importance sampling estimate. Because of its simplicity, the importance sampling estimate in Eq. (6) remains very commonly used. We describe here a cheap and simple way to improve it that was already outlined, but not implemented, by Cremer et al. (2018).

Rather than reusing the trained encoder $g_{\boldsymbol{\gamma}^*}$, we can fit a *new encoder* $g_{\tilde{\boldsymbol{\gamma}}}$ by minimising the Kullback-Leibler divergence between $q_{\boldsymbol{\gamma}}(\mathbf{z}|\tilde{\mathbf{x}})$ and $p_{\boldsymbol{\theta}}(\mathbf{z}|\tilde{\mathbf{x}})$ with respect to $\boldsymbol{\gamma}$, which is equivalent to finding

$$\tilde{\boldsymbol{\gamma}} \in \operatorname*{arg\,min}_{\boldsymbol{\gamma} \in \Gamma} \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\gamma}}(\mathbf{z}|\tilde{\mathbf{x}})} \left[\log \frac{p_{\boldsymbol{\theta}^*}(\tilde{\mathbf{x}}, \mathbf{z})}{q_{\boldsymbol{\gamma}}(\mathbf{z}|\tilde{\mathbf{x}})} \right]. \tag{7}$$

This new encoder $g_{\tilde{\boldsymbol{\gamma}}}$ will lead to a proposal $q_{\tilde{\boldsymbol{\gamma}}}(\mathbf{z}|\tilde{\mathbf{x}})$ that is closer (in the Kullback-Leibler sense) to the optimal one $p_{\boldsymbol{\theta}^*}(\mathbf{z}|\tilde{\mathbf{x}})$. Of course, if a single new data point is considered, fitting a new encoder is unnecessary and one can simply perform local variational inference, as suggested by Cremer et al. (2018). The main advantage of the refitting is evident when we are dealing with many out-of-sample data points $\tilde{\mathbf{x}}_1, ..., \tilde{\mathbf{x}}_N$, and that the optimisation problem naturally becomes

$$\tilde{\boldsymbol{\gamma}} \in \operatorname*{arg\,min}_{\boldsymbol{\gamma} \in \Gamma} \sum_{i=1}^{N} \mathbb{E}_{\mathbf{z} \sim q_{\boldsymbol{\gamma}}(\mathbf{z}|\tilde{\mathbf{x}}_i)} \left[\log \frac{p_{\boldsymbol{\theta}^*}(\tilde{\mathbf{x}}_i, \mathbf{z})}{q_{\boldsymbol{\gamma}}(\mathbf{z}|\tilde{\mathbf{x}}_i)} \right]. \tag{8}$$

Conveniently, this objective function is just the ELBO of the new data, and can be optimised quickly by initialising the encoder using $g_{\boldsymbol{\gamma}^*}$ and performing a few passes of stochastic gradient descent through the out-of-sample data.

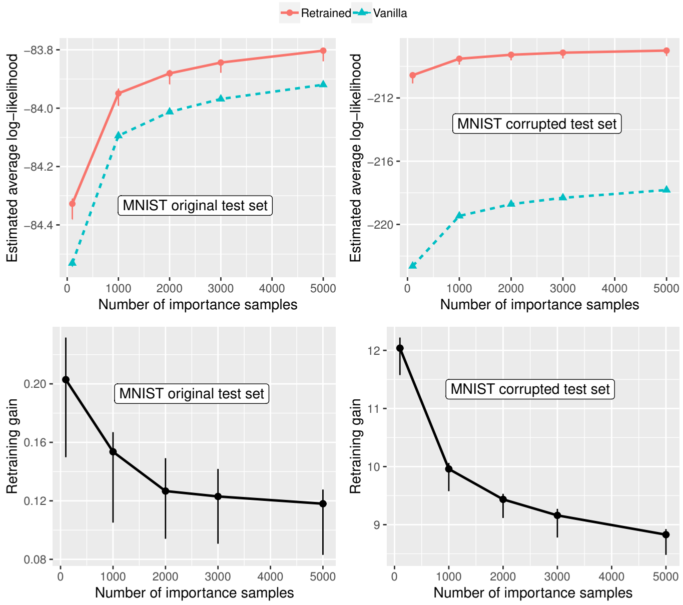

Figure 1: *Top:* Estimates of the average log-likelihood obtained with importance sampling using the original vanilla encoder $g_{\boldsymbol{\gamma}^*}$ and the retrained one $g_{\tilde{\boldsymbol{\gamma}}}$ (median and quartiles over 10 replications). *Bottom:* Difference between the estimates. As predicted by the theory of Burda et al. (2016), the gap is decreasing. However, its very slow decay motivates the refitting of the encoder.

But isn't it cheating to refit with out-of-sample data? The short answer is *no, because the encoder is not part of the generative model.* More precisely, if the likelihood $p_{\boldsymbol{\theta}^*}(\tilde{\mathbf{x}})$ is the quantity of interest, then, since this quantity does not involve γ, we are free to choose the value of γ as we wish. It is then reasonable to choose a value that will lead to a good estimate of the quantity of interest $p_{\boldsymbol{\theta}^*}(\tilde{\mathbf{x}})$, which is the goal of our refitting process. In particular, if the goal is model comparison, it is of paramount importance to obtain a good estimate of the test likelihood, irregardless of the quality of the encoder. Note, however, that if we wished to assess the quality of the couple encoder/decoder rather than of the model (for example, by looking at the test value of the IWAE objective), refitting the encoder on a test set would be proscribed.

3 Experiments

We train a VAE on the static binarisation of MNIST, using the convolutional architecture of Salimans et al. (2015). Training is done using the Adam optimiser (Kingma and Ba, 2014) and the gradient estimates of Roeder et al. (2017). We then use the trained VAE to evaluate the log-likelihood of the MNIST test set, and a slightly corrupted version. For the corrupted version, we simply take the original test set, and change the value of 10 pixels chosen uniformly at random in each image. The log-likelihood is estimated using the IWAE objective. Since this is a stochastic lower bound of the true log-likelihood, a higher objective will mean a more accurate estimation. As proposal, we use either the original encoder or a refitted one obtained by performing 10 passes over the test data.

Results for various numbers of samples are provided in Fig. 1. Even when several thousands of samples are used, the refitting procedure significantly improves the accuracy of the log-likelihood estimate, especially in the slightly corrupted case.

4 Conclusion

The most commonly used performance metric of VAEs and related models is the test log-likelihood estimate obtained using $5\,000$ importance samples and an encoder fit on training data. In spite of the known shortcomings (Wu et al., 2017; Cremer et al., 2018; Nowozin, 2018) of this approach, which is also confirmed by our experiments, it is likely that this estimate will still be used frequently in practice, because of its ease of implementation. We suggest that the refitting process described in this note can play the role of a convenient and easy improvement to this popular estimate.

References

Y. Burda, R. Grosse, and R. Salakhutdinov. Importance weighted autoencoders. *International Conference on Learning Representations*, 2016.

C. Cremer, X. Li, and D. Duvenaud. Inference suboptimality in variational autoencoders. *arXiv preprint arXiv:1801.03558*, 2018.

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *Proceedings of the International Conference on Learning Representations*, 2014.

D. P. Kingma and M. Welling. Auto-encoding variational Bayes. In *Proceedings of the International Conference on Learning Representations*, 2014.

S. Nowozin. Debiasing evidence approximations: On importance-weighted autoencoders and jackknife variational inference. In *International Conference on Learning Representations*, 2018.

T. Rainforth, A. R. Kosiorek, T. A. Le, C. J. Maddison, M. Igl, F. Wood, and Y. W. Teh. Tighter variational bounds are not necessarily better. In *International Conference on Machine Learning*, 2018.

D. Rezende, S. M. A. Eslami, S. Mohamed, P. Battaglia, M. Jaderberg, and N. Heess. Unsupervised learning of 3D structure from images. In *Advances in Neural Information Processing Systems 29*, pages 4996–5004. 2016.

D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1278–1286, 2014.

G. Roeder, Y. Wu, and D. Duvenaud. Sticking the landing: Simple, lower-variance gradient estimators for variational inference. In *Advances in Neural Information Processing Systems 30*, pages 6928–6937. 2017.

T. Salimans, D. P. Kingma, and M. Welling. Markov chain Monte Carlo and variational inference: Bridging the gap. In *International Conference on Machine Learning*, pages 1218–1226, 2015.

G. Tucker, D. Lawson, S. Gu, and C. J. Maddison. Doubly reparameterized gradient estimators for Monte Carlo objectives. *arXiv preprint arXiv:1810.04152*, 2018.

Y. Wu, Y. Burda, R. Salakhutdinov, and R. Grosse. On the quantitative analysis of decoder-based generative models. *Proceedings of the International Conference on Learning Representations*, 2017.