LOFT - Stable Training of Normalizing Flows for Variational Inference

Daniel Andrade[1]

[1]Education and Research Center for Artificial Intelligence and Data Innovation, Hiroshima University, Hiroshima, Japan

Abstract

Variational Inference (VI) with Normalizing Flows (NFs) is an increasingly popular alternative to MCMC methods. However, despite recent progress on stabilizing the variance of stochastic gradient descent during training, we observe that convergence is still difficult to achieve in practice. In particular, if the target distribution's dimension is high or exhibits fat tails, convergence of NFs fail and only the much simpler Gaussian mean-field VI converges. As a remedy, we introduce the log soft extension (LOFT) layer, which can effectively restrain the samples of NFs to lie in a reasonable range. For various different target distributions with high-dimensions or fat tails, we observe that LOFT enables successful training of NFs that was previously not possible. Moreover, the computational overhead of the LOFT layer is only marginal. Therefore, we expect that LOFT becomes a new standard tool for training deep NFs for Bayesian inference.

1 INTRODUCTION

During the past decade, variational inference (VI) has been gaining increased attention also in the statistics community [Blei et al., 2017]. Thanks to its computational efficiency, VI can provide useful approximations to high-dimensional posterior distribution that are computationally too expensive using MCMC methods. However, simple variational distributions, like the Gaussian distribution, can lead to arbitrarily bad approximations [Zhang et al., 2018].

As a promising method for increasing the flexibility of the VI approximation, normalizing flows (NFs) have been proposed [Papamakarios et al., 2021]. Unfortunately, for high dimensional posterior distributions VI with NFs is known to suffer from gradient estimates with high variance [Dhaka et al., 2021].

Recently, two methods for stabilizing training and mitigating the high variance have been proposed: ActNorm and path gradients. ActNorm [Kingma and Dhariwal, 2018] is an in-between-layer normalization similar to batch normalization, but with the difference that the statistics (mean and standard deviation that are used for normalization) are estimated only at the beginning of the training. On the other hand, path gradients [Roeder et al., 2017, Vaitl et al., 2022] remove the score term from the gradient estimation of the Kullback-Leibler (KL)-divergence, which can sometimes lead to considerably lower variance of the gradient estimates.

Unfortunately, we show here that for various different target distributions, ActNorm and path gradients are still insufficient to stabilize the training of deep NFs. We note that control variates, as proposed in [Ranganath et al., 2014], could in principle also be applied to NFs. However, in practice, due to the high number of parameters of NFs, such control variates are not computationally feasible.

As a remedy, we introduce the log soft extension (LOFT) layer that helps to restrain samples from the normalizing flows. We demonstrate that when using LOFT, the NFs' samples used for gradient estimation can exhibit considerably lower variance, which consequently ensures convergence even for high-dimensional target distributions with fat tails. Our experiments show that for various challenging target distributions LOFT enables successful training of NFs that was previously not possible.

2 LIMITATIONS OF EXISTING METHODS FOR VARIANCE STABILIZATION

Following the works of [Dhaka et al., 2021], we consider here the reverse Kullback-Leibler (KL) divergence with masked affine flows (Real NVP, [Dinh et al., 2016]), since

Accepted for the 6th Workshop on Tractable Probabilistic Modeling at UAI (TPM 2023).

other choices exhibit too much instability during training. In detail, variational inference with the reverse KL divergence minimizes

$$\mathrm{KL}(q_{\boldsymbol{\vartheta}}||p_*) = \mathbb{E}_{q_{\boldsymbol{\vartheta}}}\left[\log\left(\frac{q_{\boldsymbol{\vartheta}}(\boldsymbol{\theta})}{p_*(\boldsymbol{\theta})}\right)\right], \tag{1}$$

where $q_{\boldsymbol{\vartheta}}$ denotes the variational approximation to the target density p_*. In Bayesian statistics, $p_*(\boldsymbol{\theta})$ corresponds to the posterior distribution $p(\boldsymbol{\theta}|D)$.[1] In particular, here, we assume that $q_{\boldsymbol{\vartheta}}$ is specified by a Real NVP with a Gaussian base distribution $q_0(\boldsymbol{u})$ and a one-to-one transformation $t(\boldsymbol{u};\boldsymbol{\vartheta})$. For finding the optimal parameters $\boldsymbol{\vartheta}$, we use gradient descent, with samples from $q_{\boldsymbol{\vartheta}}$ to get an unbiased estimate of $\nabla_{\boldsymbol{\vartheta}}\mathrm{KL}(q_{\boldsymbol{\vartheta}}||p_*)$.[2] Furthermore, for reducing the variance of the gradient estimates, we employ ActNorm [Kingma and Dhariwal, 2018] and path gradients [Vaitl et al., 2022].

As can be seen in the top of Figure 1, the variance of the estimate of Equation 1 can be considerable, and, as a consequence, gradient estimates become unstable. In particular, we observe that for high-dimensional distributions, or distributions with fat tails, the variance of the ELBO (evidence lower bound) increases up to the degree that training becomes impossible.

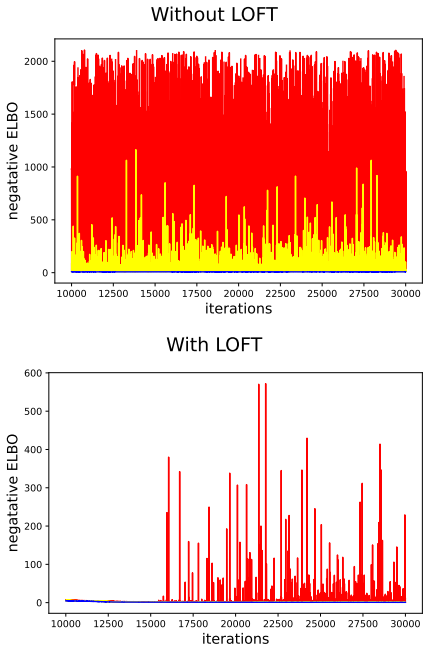

Figure 1: Shows the maximum value (red), 99% (yellow) and 75% (blue) quantiles of the negative ELBO (256 samples) at each training iteration. Here, target distribution p_* is the Multivariate T-Distribution with $d = 10$ as described in Section 4.1.

3 PROPOSED METHOD

In order to soften the impact of (unusual) large samples from $q_{\boldsymbol{\vartheta}}$, we propose the following log soft extension (LOFT) layer:

$$f(\theta) = \begin{cases} t + \log(\theta - t + 1) & \text{if } \theta \geq t, \\ -t - \log(-\theta - t + 1) & \text{if } \theta \leq -t, \\ \theta & \text{else.} \end{cases}$$

$$= \mathrm{sign}(\theta)\Big(\log\big(\max(|\theta| - t, 0) + 1\big) + \min(|\theta|, t)\Big).$$

The function is shown in Figure 2: within the range $[-t, t]$ the layer performs an identity mapping, and outside the range, the absolute value of the function grows only logarithmically. t is a fixed user-specified parameter. In particular, for Bayesian statistics, we suggest to set t as high as necessary such that most of the mass of $p(\boldsymbol{\theta}|D)$ is expected to be covered by $[-t, t]^d$. We suggest, to interleave the NFs with LOFT layers, similar to the usage of ActNorm.

Note that LOFT is a one-to-one function and

$$f^{-1}(z) = \mathrm{sign}(z)\Big(\exp\big(\max(|z| - t, 0)\big) - 1 + \min(|z|, t)\Big),$$

$$\log\Big(\frac{\partial}{\partial\theta} f(\theta)\Big) = -\log\big(\max(|\theta| - t, 0) + 1\big).$$

Therefore, all necessary calculations can be expressed using only computationally efficient elementary operations (without if-clauses).

4 EXPERIMENTS

For all experiments, we use Adam [Kingma and Ba, 2015] with a learning rate of 10^{-5} to optimize the ELBO. We run training for $3 \cdot 10^4$ iterations, where for the initial 10^4 iterations we use annealing with a linearly increasing temperature. In order to decrease the variance of the ELBO-estimate, we use path gradients as described in Roeder et al. [2017], Vaitl et al. [2022]. In preliminary experiments, we

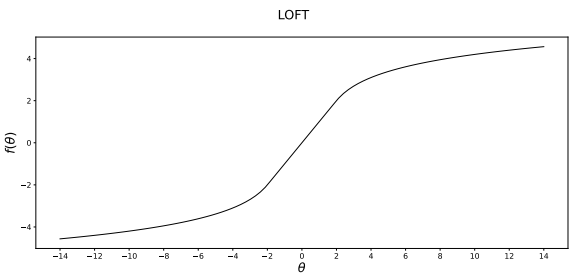

Figure 2: LOFT function. Note that for a vector $\boldsymbol{\theta}$ the LOFT function is applied element-wise.

[1] $\boldsymbol{\theta}$ are the parameters of the Bayesian model and D is the data.

[2] With the help of the reparameterization trick, details see e.g. [Papamakarios et al., 2021].

confirmed that (except in the initial phase) this helps to reduce the variance of the gradient estimates.

For the normalizing flows, we use 64 masked affine flows (Real NVP, [Dinh et al., 2016]), where after each flow we additionally add an ActNorm layer [Kingma and Dhariwal, 2018]. Again, we confirmed in preliminary experiments that the ActNorm layer helped to reduce variance. The proposed method uses an additional LOFT layer after each ActNorm layer. For all experiments, we set $t := 100$. For the implementation, we build upon the Python package Normflows [Stimper et al., 2023]. All experiments were conducted on an Nvidia DGX2 with double precision.

4.1 TARGET DISTRIBUTIONS

For evaluation, we use four different target distribution, for which the log normalization constant is known.

Funnel Distribution The funnel distribution, as introduced in [Neal, 2003], is given by

$$p_*(\theta_1) = N(0, 9), \quad \forall j \in 2, \dots, d : p(\theta_j|\theta_1) = N(0, e^{\theta_1}) \,.$$

The target distribution is $p_*(\boldsymbol{\theta})$ with $\boldsymbol{\theta} := (\theta_1, \theta_2, \dots, \theta_d) \in \mathbb{R}^d$.

Multivariate T-Distribution The multivariate student-t distribution with mean $\mathbf{0}$, degrees of freedom ν, and scale matrix Σ is given by

$$p_*(\boldsymbol{\theta}) = \frac{\Gamma((\nu+d)/2)}{\Gamma(\nu/2)(\nu\pi)^{d/2}|\Sigma|^{1/2}} \Big(1 + \frac{1}{\nu}\boldsymbol{\theta}^T\Sigma^{-1}\boldsymbol{\theta}\Big)^{-(\nu+d)/2},$$

where we set the scale matrix to $\Sigma_{i,j} = 0.8$, for $i \neq j$, and $\Sigma_{i,i} = 1.0$.

Multivariate Gaussian Mixture

$$p_*(\boldsymbol{\theta}) = \sum_{j=1}^{k} \frac{1}{k} N(\boldsymbol{\theta}|\boldsymbol{\mu}_j, I_d) \,,$$

where I_d denotes the identity matrix in $\mathbb{R}^{d\times d}$. Here, we set $k = 3$, and $\boldsymbol{\mu}_1 = \mathbf{4}$, $\boldsymbol{\mu}_2 = -\mathbf{4}$, and $\boldsymbol{\mu}_3 = \mathbf{0}$ (i.e. in all dimensions constant value 4, -4, and 0, respectively).

Bayesian Linear Regression We consider the following Bayesian linear regression model

$$\begin{aligned}
&\sigma^2 \sim \text{Inv-Gamma}(0.5, 0.5) \\
&\boldsymbol{\beta} \sim N(\mathbf{0}, \sigma^2 I_{d-1}) \\
&y_i \overset{i.i.d}{\sim} N(\mathbf{x}_i^T\boldsymbol{\beta}, \sigma^2) \quad \text{for } i \in \{1, \dots n\} \,.
\end{aligned}$$

The parameters are $\boldsymbol{\theta} := (\boldsymbol{\beta}, \sigma^2)$, and the target distribution is the posterior $p(\boldsymbol{\theta}|\mathbf{y}, X)$. Note that, different from before, $\boldsymbol{\theta} \in \mathbb{R}^{d-1} \otimes \mathbb{R}_+$. In order to ensure the positiveness for σ^2, we use the softplus-transformation (as suggested, for example, in Kucukelbir et al. [2017]).

Due to the conjugacy of the priors, the marginal likelihood has a closed form given by Chipman et al. [2001]:

$$\log p(\mathbf{y}|X) = \frac{\Gamma((1+n)/2)}{\Gamma(1/2)(\nu\pi)^{n/2}|\Sigma|^{1/2}} \Big(1 + \mathbf{y}^T\Sigma^{-1}\mathbf{y}\Big)^{-(1+n)/2},$$

where $\Sigma := (I_n - X(X^TX + I_d)^{-1}X^T)^{-1}$, and $X \in \mathbb{R}^{n\times d}$ contains all explanatory variables.

4.2 EVALUATION AND RESULTS

As can be seen in Figure 1 (bottom), with the help of the LOFT layer, the variance of the samples are greatly reduced. Indeed training of the normalizing flows with LOFT (NF+LOFT) always converged. However, without the LOFT layer, training was often numerically too unstable, i.e. did not converge due to the high variance of the gradients. As an ad-hoc remedy, we therefore also trained the normalizing flows with weight decay (penalty was set to 1.0) and gradient clipping (l2-norm set to 10.0) which we name NF+RC. We also compared to a mean field Gaussian approximation (Gaussian-MF).

For evaluation (after training), we use 20000 samples, and repeat each evaluation 20 times to estimate the Monte Carlo error [Koehler et al., 2009]. In Table 1, we show the ELBO of all methods for all target distributions. Furthermore, after training of each variational approximation q, we use q as proposal distribution for importance sampling to estimate the marginal likelihood (i.e. the normalization constant Z of the target distribution p_*). Table 2 shows the mean absolute error $|\log Z - \log \tilde{Z}|$, where Z and $\tilde{Z}$ denote the true and approximated marginal likelihood, respectively.

We note that the computational overhead of LOFT is marginal: training time with the LOFT layer increased only between around 10% (for $d = 1000$) and 30% (for $d \leq 100$).

5 CONCLUSIONS

For various different target distributions with high-dimensions or fat tails, we observe that LOFT enables successful training of deep NFs that was previously not possible. Notably, even ad hoc measures like weight decay and gradient clipping either did not prevent unstable gradients or led to convergence towards an inferior solution. Moreover, since the computational overhead of the LOFT layer is only marginal, we expect that LOFT becomes a new standard tool for training deep NFs for Bayesian inference.

Table 1: Evaluation of all methods in terms of ELBO (standard deviation in brackets) for $d \in \{10, 100, 1000\}$. "NA" means that a method did not converge.

		Funnel		
d	**Gaussian-MF**	**NF**	**NF+RC**	**NF+LOFT**
10	-22607.68945 (112.80378)	**-0.004** (0.00065)	-0.37008 (0.0079)	**-0.004** (0.00065)
100	-311913.65625 (1204.79443)	**-0.01529** (0.00132)	-0.82154 (0.00606)	-0.01852 (0.00141)
1000	-3022390.0 (11822.16797)	**-0.07086** (0.00247)	NA	-0.07201 (0.00334)
		Multivariate Student-t		
10	-17.72555 (0.01859)	NA	-0.92091 (0.00434)	**-0.44335** (0.02281)
100	-153.34932 (0.03154)	NA	NA	**-1.29345** (0.00552)
1000	-1450.84351 (0.16274)	NA	NA	**-432.67764** (0.29864)
		Multivariate Gaussian Mixture		
10	-113.17033 (0.11369)	-1.09873 (0.0001)	NA	**-1.09861** (1e-05)
100	-1583.33484 (0.45218)	-1.09984 (0.0003)	-1.09896 (0.00018)	**-1.09865** (0.00013)
1000	-15113.09961 (1.10571)	NA	NA	**-1.11051** (0.00075)
		Bayesian Linear Regression ($n = 2000$)		
10	-163874.53125 (334.17224)	**-5074.49707** (0.00124)	-5074.78613 (0.00808)	**-5074.49707** (0.00124)
100	-343551.25 (247.65498)	NA	NA	**-5353.60693** (0.00169)
1000	-442254432.0 (2311782.25)	NA	NA	**-95220.10156** (66.93725)
		Bayesian Linear Regression ($n = 20$)		
10	-1442.38586 (2.90839)	-55.48552 (0.00031)	-55.92112 (0.00485)	**-55.48539** (0.00038)
100	-2801.54956 (2.9594)	NA	NA	**-66.51662** (0.00561)
1000	-4344290.0 (23051.80469)	NA	NA	**-80.42844** (0.0326)

Table 2: Evaluation of all methods in terms of $|\log Z - \log \tilde{Z}|$ (i.e. difference of true and estimated normalizing constant), when using importance sampling. "NA" means that a method did not converge.

		Funnel		
d	**Gaussian-MF**	**NF**	**NF+RC**	**NF+LOFT**
10	2378.45654 (233.43074)	**0.00063** (0.00058)	0.05873 (0.0413)	**0.00063** (0.00058)
100	35628.20312 (2457.3728)	**0.00108** (0.00077)	0.34005 (0.06012)	0.00131 (0.00095)
1000	352202.96875 (25639.97656)	**0.00384** (0.00326)	NA	0.00641 (0.01088)
		Multivariate Student-t		
10	11.95926 (1.37316)	NA	0.45397 (0.30841)	**0.14678** (0.11084)
100	127.72206 (4.10065)	NA	15.66333 (1.57771)	**0.92498** (0.02456)
1000	1366.32544 (6.39189)	NA	178.99582 (7.55553)	**159.49564** (17.38598)
		Multivariate Gaussian Mixture		
10	62.47577 (2.58455)	1.09863 (0.00011)	NA	**1.09862** (1e-05)
100	1365.84741 (12.62567)	1.0986 (0.00034)	1.09862 (0.00017)	**1.09859** (9e-05)
1000	14414.19824 (37.7167)	NA	119743.08594 (16996.94141)	**1.09838** (0.00094)
		Bayesian Linear Regression ($n = 2000$)		
10	65232.39453 (3233.09131)	**0.00056** (0.00035)	0.00627 (0.00478)	**0.00056** (0.00035)
100	251705.39062 (3288.46167)	NA	NA	**0.00129** (0.00078)
1000	32286436.0 (4261146.0)	NA	NA	**49792.94141** (1364.07861)
		Bayesian Linear Regression ($n = 20$)		
10	567.07452 (31.58895)	**0.00023** (0.00017)	0.06655 (0.09409)	0.00025 (0.00018)
100	1704.42737 (49.23442)	NA	111.10826 (6.41515)	**0.01291** (0.01942)
1000	301136.15625 (42675.53516)	NA	NA	**0.83201** (0.43196)

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number 22K11934.

References

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

Hugh Chipman, Edward I George, Robert E McCulloch, Merlise Clyde, Dean P Foster, and Robert A Stine. The practical implementation of bayesian model selection. *Lecture Notes-Monograph Series*, pages 65–134, 2001.

Akash Kumar Dhaka, Alejandro Catalina, Manushi Welandawe, Michael R Andersen, Jonathan Huggins, and Aki Vehtari. Challenges and opportunities in high dimensional variational inference. *Advances in Neural Information Processing Systems*, 34:7787–7798, 2021.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in neural information processing systems*, 31, 2018.

Elizabeth Koehler, Elizabeth Brown, and Sebastien J-PA Haneuse. On the assessment of monte carlo error in simulation-based statistical analyses. *The American Statistician*, 63(2):155–162, 2009.

Alp Kucukelbir, Dustin Tran, Rajesh Ranganath, Andrew Gelman, and David M Blei. Automatic differentiation variational inference. *Journal of machine learning research*, 2017.

Radford M Neal. Slice sampling. *The annals of statistics*, 31(3):705–767, 2003.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021.

Rajesh Ranganath, Sean Gerrish, and David Blei. Black box variational inference. In *Artificial intelligence and statistics*, pages 814–822. PMLR, 2014.

Geoffrey Roeder, Yuhuai Wu, and David K Duvenaud. Sticking the landing: Simple, lower-variance gradient estimators for variational inference. *Advances in Neural Information Processing Systems*, 30, 2017.

Vincent Stimper, David Liu, Andrew Campbell, Vincent Berenz, Lukas Ryll, Bernhard Schölkopf, and José Miguel Hernández-Lobato. normflows: A PyTorch Package for Normalizing Flows. *arXiv preprint arXiv:2302.12014*, 2023.

Lorenz Vaitl, Kim A Nicoli, Shinichi Nakajima, and Pan Kessel. Gradients should stay on path: better estimators of the reverse-and forward kl divergence for normalizing flows. *Machine Learning: Science and Technology*, 3(4): 045006, 2022.

Cheng Zhang, Judith Bütepage, Hedvig Kjellström, and Stephan Mandt. Advances in variational inference. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):2008–2026, 2018.