2nd Symposium on Advances in Approximate Bayesian Inference, 2019 1–9

Efficient Inference Amortization in Graphical Models using Structured Continuous Conditional Normalizing Flows

Christian Weilbach WEILBACH@CS.UBC.CA
Boyan Beronov BERONOV@CS.UBC.CA
William Harvey WSGH@CS.UBC.CA
Frank Wood FWOOD@CS.UBC.CA
University of British Columbia

1. Overview

We introduce a more efficient neural architecture for amortized inference (Gershman, 2014; Ritchie et al., 2016), which combines continuous (Grathwohl et al., 2018) and conditional (Chen et al., 2019) normalizing flows using a principled choice of structure. Our flow derives its sparsity pattern from the minimally faithful inverse of its underlying graphical model (Webb et al., 2018). We find that this factorization reduces the necessary numbers both of parameters in the neural network and of adaptive integration steps in the ODE solver. Consequently, the throughput at training time and inference time is increased, without decreasing performance in comparison to unconstrained flows. By expressing the structural inversion and the flow construction as compilation passes of a probabilistic programming language, we demonstrate their applicability to the stochastic inversion of realistic models such as convolutional neural networks (CNN).

Our automated pipeline consists of three program transformations, as illustrated in Figure 1: First, a formal specification of a generative process is translated into a graphical model, and its minimally faithful inverse structure is computed as described in Section 2. Subsequently, the latter acts as the sparsity pattern for the novel neural network architecture introduced in Section 3. Finally, the resulting flow is trained with a novel symmetrized KL loss, as summarized in Section 4.

2. Faithful Model Inversion

Given a static graphical model, we apply the faithful inversion algorithm of Webb et al. (2018), and obtain a correct dependence structure for the inverse model $p(z|x)$, which maps from observations x to latents z. In particular, this algorithm computes a structure with minimal number of moralizing edges, which are required to capture all possible correlations in the posterior. As an example, Panel (3) in Figure 1 shows the minimally faithful inverse of the graphical model in Panel (2).

© C. Weilbach, B. Beronov, W. Harvey & F. Wood.

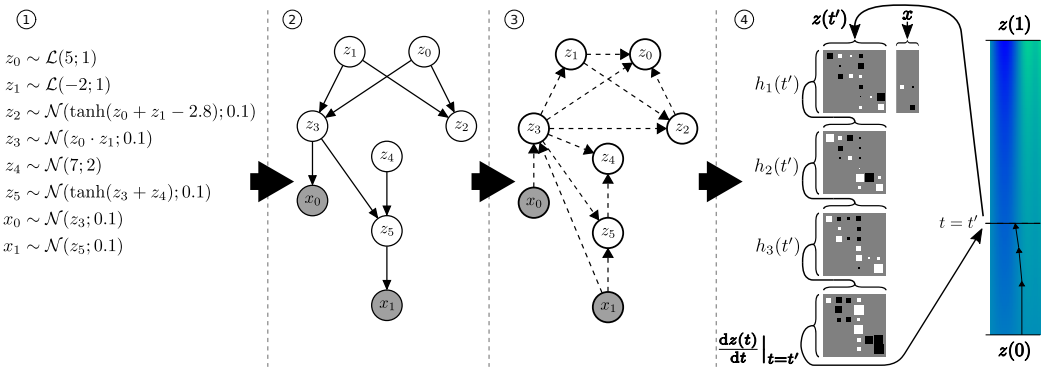


Figure 1: A generative model (1) is compiled into a graphical model (2) and stochastically inverted (3). This structure is translated into the sparsity pattern of the neural network (4), which approximates the posterior $p(z_{0,\dots,4}|x_{0,1})$ as a continuous normalizing flow under the control input x. The flow network's architecture is depicted using Hinton diagrams (Hinton and Shallice, 1991) of its layer-wise weight matrices – with color and size denoting sign and magnitude, and columns and rows corresponding to inputs and outputs of layers. For clarity, augmenting dimensions are not shown.

3. Structured Normalizing Flows

3.1. Amortized Inference with Continuous Normalizing Flows

Amortized inference techniques (Gershman, 2014; Ritchie et al., 2016) yield efficient posterior approximations as a result of training function approximators on losses defined using the generative model and training data, e.g., the variational evidence lower bound (Blei et al., 2016; Kingma and Welling, 2013). The general framework used here for inference amortization is a neural ordinary differential equation (ODE) system (Chen et al., 2018), a differentiable deterministic transformation from a reference distribution q^0 to the desired target density $q_\Phi(\cdot \mid x)$. This transformation is parametrized as a on latent particles z,

$$\frac{\mathrm{d}}{\mathrm{d}t} z_t = f_\Phi(z_t, t, x). \tag{1}$$

where conditioning is achieved by providing x as a constant control input to the neural network f_Φ. The numerical computation at inference time is then performed by a standard ODE solver, integrating independent particle trajectories along the dynamics in Equation (1), from initial conditions $z_0 \sim q^0$ to approximate posterior samples $z_1 \sim q_\Phi(\cdot \mid x)$. In order to obtain a normalized distribution at the end of the flow, the log-probability of each particle must also be integrated alongside the particle dynamics as

$$\frac{\partial}{\partial t} \ln q_\Phi(z, t) = -\nabla_z \cdot f_\Phi(z, t, x) \ , \tag{2}$$

where ∇_z denotes the gradient operator in the latent space. This divergence term is equivalent to the trace of the Jacobian of f_Φ.

There are two main algorithmic advantages of this approach: its intrinsic parallelism between independent particles, and the trivial reversibility of the flow transformation using

the same integrator in opposite direction. Recently, such flows have also been applied to graph neural networks as a form of continuous message passing (Deng et al., 2019; Liu et al., 2019). Our work differs from such literature chiefly in two ways: we target inference amortization instead of density estimation, and our flows represent a global continuous message passing dynamics on the sparse inverse model structure.

3.2. Sparse Neural ODE

In order to constrain the architecture of the flow network f_Φ to respect the necessary statistical independence structure, the weight matrix of each layer h_{Φ_l} is masked with the adjacency H of the minimally faithful inverted graphical model, i.e., the output reads

$$\begin{aligned} f_\Phi(z, t, x) &= \big(h_{\Phi_L}(\cdot, t) \circ \cdots \circ h_{\Phi_1}(\cdot, t)\big)(z \oplus x) \\ h_{\Phi_l}(\hat{z}, t) &= \sigma\big\{(W_l \odot H)\hat{z} \odot \eta_{l,1}(t)\big\} + b_l \odot \eta_{l,2}(t) \end{aligned} \tag{3}$$

Here the column $\big\{(h_{\Phi_l}(\hat{z}, \cdot))_i\big\}_l$ of activations across layers l corresponds to a node i in the graphical model, σ is the activation function tanh, b is a bias, and $\eta_{l,\cdot}$ are time dependent linear gating functions modelling time dependencies of the flow as in Chen et al. (2018). While our architecture captures the global statistical structure, we have not yet explored inversion of individual link functions as in Tavares and Lezama (2016). Optionally, we introduce local nuisance variables to increase the latent space dimension of each random variable, following similar reasoning to Dupont et al. (2019). In our experiments, we found $L = 3$ hidden layers to provide enough over-parametrization. A simplified version of this architecture is shown in Panel (4) of Figure 1.

4. Symmetric KL

Our optimization objective is the symmetrized Kullback-Leibler divergence in expectation over training data,

$$\begin{aligned} \mathcal{L}\left[q_\Phi\right](\mathcal{X}) &= \frac{1}{2}\,\mathbb{E}_{x\sim\mathcal{X}}\Big[\underbrace{\mathcal{D}_{\mathrm{KL}}\{p(\cdot \mid x)\,||\,q_\Phi(\cdot \mid x)\}}_{\text{forward KL}} + \underbrace{\mathcal{D}_{\mathrm{KL}}\{q_\Phi(\cdot \mid x)\,||\,p(\cdot \mid x)\}}_{\text{reverse KL}}\Big] \\ &= \frac{1}{2}\mathbb{E}_{x\sim\mathcal{X}}\left[\mathbb{E}_{z\sim p(\cdot|x)}\left[\ln\frac{p(z,x)}{q_\Phi(z \mid x)}\right] + \mathbb{E}_{z\sim q_\Phi(\cdot|x)}\left[\ln\frac{q_\Phi(z \mid x)}{p(z,x)}\right]\right]. \end{aligned} \tag{4}$$

While the forward KL term measures the quality of density estimation on the support of the true posterior, the reverse KL term incentivizes samples from q_Φ to behave similarly to the latter. Efficient estimation of this objective is possible in this setting, because the joint model is available and the variational posterior q_Φ is reparametrized. Importantly, in contrast to expected forward or reverse KL alone, $\mathcal{L}\left[q_\Phi\right]$ does not contain the unknown constant factor $\mathbb{E}_{x\sim\mathcal{X}} \ln p(x)$. In the experiments described below, we uniformly find a significant performance improvement over using only the forward or reverse KL for training.

5. Experiments

5.1. Arithmetic Circuit

A quantitative comparison of the minimally faithful inversion structure against three baselines is shown in Figure 2, measuring the objective in Equation (4) on the arithmetic circuit example from Figure 1. Aside from using the same algorithm and hyperparameters, the different architectures are made comparable by choosing similar numbers of dimensions for the latent spaces of the flows: FFJORD (Grathwohl et al., 2018) transforms the original 6D latent space into the flow space using a fully connected layer, while for the other architectures we augment each original latent dimension by 10 additional dimensions. As a result, FFJORD (64 dimensions, 17801 parameters) and the flows with full connectivity (66 dimensions, 18679 parameters) and minimally faithful inverse structure (66 dimensions, 7725 parameters) achieve competitive performances. In addition to using significantly fewer parameters, our sparsity structure trains faster in the beginning, suggesting a more appropriate inductive bias. The importance of faithful inversion is corroborated by a control experiment, which differs only in its randomized sparsity structure (66 dimensions, 7725 parameters), and performs poorly, suffering from early saturation and high variance.

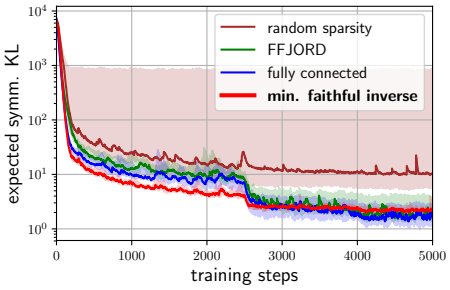


Figure 2: Effect of flow architecture on the training loss (expected symmetrized KL) for the experiment in Figure 1. Depicted are the means and the 1σ confidence intervals for 10 runs each, smoothed in time.

5.2. Objective Function

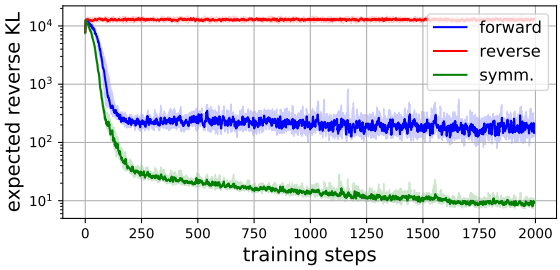
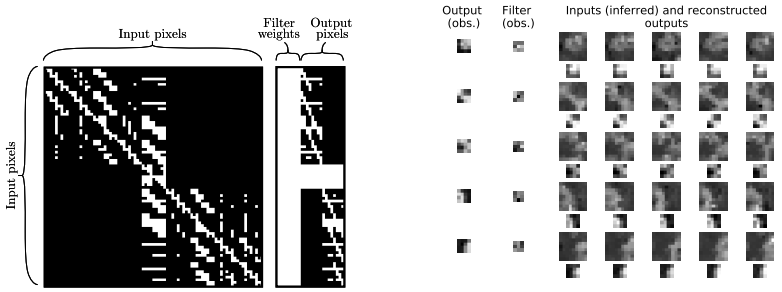


Figure 3: Effect of the choice of objective function (expected forward/reverse/symmetrized KL), measured in terms of the expected reverse KL (ELBO) for the experiment in Figure 1. By accounting for both divergences, and thereby combining mode-seeking and mass-seeking behaviour, the symmetrized loss provides a stronger learning signal in general. In this example, the validation loss improves by more than an order of magnitude. We plot the median and a band of one standard deviation over 10 runs.

Figure 3 shows a comparison of the different losses described in Section 4. The reverse KL-based loss was found to be capable of training simpler models, such as small Gaussian

state space models. However, it had consistently higher variance than the forward KL and was not at all sufficient for training on the arithmetic circuit we consider, as Figure 3 shows. The forward KL, the standard loss introduced with CNFs (Grathwohl et al., 2018), provides a learning signal on the task, but quickly saturates with a reversed KL of about 100 nats. The symmetrized KL, on the other hand, learns faster from the start and keeps improving to below 10 nats. This is a crucial improvement, since the forward KL only optimizes q to be a density estimator for $p(z|x)$, while the reverse KL optimizes the sampling behavior of q as well. Our experiment shows that such a CNF can only be trained with the symmetrized KL. For this run we have used an augmentation of 5 dimensions for each latent variable, all the other parameters were the same as in the previous result. The benefits of the symmetrizeded loss wereconsistent over all our experiments.

5.3. Deconvolution



(a) (b)

Figure 4: (a) Adjacency matrix of the minimally faithful inverse structure for a 2D convolution, using the dimension convention of Figure 1 and black/white for 0/1. (b) Examples of stochastic deconvolution, trained as a flow with the sparsity pattern in (4a). Each row conditions on an output (4×4) and a filter (3×3) to infer corresponding inputs (9×9). Posteriors are visualized using 5 samples of the input and the reconstructed outputs.

As an example for a more challenging application, Figure 4 portrays 2D deconvolution, interpreted as amortized inference for the generative process of image convolution. To obtain output pixels (4×4), the generative model samples each of the filter weights (3×3) from a standard normal prior and calculates the forward convolution on an image patch (9×9) with stride 2 and no padding. The minimally faithful inversion structure in Figure 4a indicates all statistical dependencies: across (inferred) input pixels, of inputs on filter weights, and of input pixels on their outputs. For example, pixels in the middle of the input patch visibly depend on all output values. The inference artifact is trained on randomly cropped real image patches from the MNIST digit classification dataset, and amortizes over all possible convolutional filters of this shape. It should be noted that in contrast to usual deconvolutional architectures, this stochastic inverse function is trained without explicit weight sharing. Finally, we perform a qualitative consistency check in Figure 4b, by reconstructing outputs from samples of the approximate posterior.

Acknowledgments

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), the Canada CIFAR AI Chairs Program, Compute Canada, Intel, and DARPA under its D3M and LWLL programs.

References

Jens Behrmann, Will Grathwohl, Ricky T. Q. Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In Chaudhuri and Salakhutdinov (2019), pages 573–582.

Eli Bingham, Jonathan P. Chen, Martin Jankowiak, Fritz Obermeyer, Neeraj Pradhan, Theofanis Karaletsos, Rohit Singh, Paul Szerlip, Paul Horsfall, and Noah D. Goodman. Pyro: Deep Universal Probabilistic Programming. *Journal of Machine Learning Research*, 2018.

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738.

David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *CoRR*, abs/1601.00670, 2016.

Jörg Bornschein and Yoshua Bengio. Reweighted wake-sleep. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Michael M. Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: Going beyond euclidean data. *IEEE Signal Process. Mag.*, 34 (4):18–42, 2017. doi: 10.1109/MSP.2017.2693418.

Kamalika Chaudhuri and Ruslan Salakhutdinov, editors. *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, 2019. PMLR.

Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 6572–6583, 2018.

Xinshi Chen, Hanjun Dai, and Le Song. Particle flow bayes' rule. In Chaudhuri and Salakhutdinov (2019), pages 1022–1031.

Marco Ciccone, Marco Gallieri, Jonathan Masci, Christian Osendorfer, and Faustino J. Gomez. Nais-net: Stable deep networks from non-autonomous differential equations. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 3029–3039, 2018.

Zhiwei Deng, Megha Nawhal, Lili Meng, and Greg Mori. Continuous graph flow for flexible density estimation. *CoRR*, abs/1908.02436, 2019.

Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *CoRR*, abs/1605.09782, 2016.

Arnaud Doucet and Adam M Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of nonlinear filtering*, 12(656-704):3, 2009.

Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *CoRR*, abs/1603.07285, 2016.

Emilien Dupont, Arnaud Doucet, and Yee Whye Teh. Augmented neural odes. *CoRR*, abs/1904.01681, 2019.

Thomas Elsken, Jan Hendrik Metzen, and Frank Hutter. Neural architecture search: A survey. *CoRR*, abs/1808.05377, 2018.

Goodman Noah D Gershman, Samuel J. Amortized Inference in Probabilistic Reasoning. page 6, 2014.

Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David K. Duvenaud. FFJORD: free-form continuous dynamics for scalable reversible generative models. *CoRR*, abs/1810.01367, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society, 2016. ISBN 978-1-4673-8851-1. doi: 10.1109/CVPR.2016.90.

Pashupati Hegde, Markus Heinonen, Harri Lähdesmäki, and Samuel Kaski. Deep learning with differential gaussian process flows. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pages 1812–1821. PMLR, 2019.

Geoffrey E. Hinton and Tim Shallice. Lesioning an attractor network: Investigations of acquired dyslexia. *Psychological Review*, 98(1):74–95, 1991. ISSN 1939-1471(Electronic),0033-295X(Print). doi: 10.1037/0033-295X.98.1.74.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *CoRR*, abs/1812.04948, 2018.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013.

Diederik P. Kingma, Tim Salimans, Rafal Józefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improving variational autoencoders with inverse autoregressive flow. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4736–4744, 2016.

Daphne Koller and Nir Friedman. *Probabilistic Graphical Models - Principles and Techniques*. MIT Press, 2009. ISBN 978-0-262-01319-2.

Tuan Anh Le, Atılım Güneş Baydin, and Frank Wood. Inference compilation and universal probabilistic programming. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 54 of *Proceedings of Machine Learning Research*, pages 1338–1348, Fort Lauderdale, FL, USA, 2017. PMLR.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998. doi: 10.1109/5.726791.

Yann Lecun and Corinna Cortes. The MNIST database of handwritten digits. 1998.

Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton. Deep learning. *Nature*, 521(7553): 436–444, 2015. doi: 10.1038/nature14539.

Jenny Liu, Aviral Kumar, Jimmy Ba, Jamie Kiros, and Kevin Swersky. Graph normalizing flows. *CoRR*, abs/1905.13177, 2019.

Behnam Neyshabur, Zhiyuan Li, Srinadh Bhojanapalli, Yann LeCun, and Nathan Srebro. The role of over-parametrization in generalization of neural networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Frank Nielsen. A family of statistical symmetric divergences based on Jensen's inequality, 2010.

Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 2016. doi: 10.23915/distill.00003.

Brooks Paige and Frank D. Wood. Inference networks for sequential monte carlo in graphical models. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pages 3040–3049, 2016.

Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. In Francis R. Bach and David M. Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1530–1538. JMLR.org, 2015.

Daniel Ritchie, Paul Horsfall, and Noah D. Goodman. Deep Amortized Inference for Probabilistic Programs. *arXiv:1610.05735 [cs, stat]*, October 2016. arXiv: 1610.05735.

Lars Ruthotto and Eldad Haber. Deep neural networks motivated by partial differential equations. *CoRR*, abs/1804.04272, 2018.

Zenna Tavares and Armando Solar Lezama. Parametric inverse simulation. 2016.

David Tolpin, Jan Willem van de Meent, Hongseok Yang, and Frank Wood. Design and implementation of probabilistic programming language anglican. *arXiv preprint arXiv:1608.05263*, 2016.

Brian L Trippe and Richard E Turner. Conditional Density Estimation with Bayesian Normalising Flows, 2018.

Jan-Willem van de Meent, Brooks Paige, Hongseok Yang, and Frank Wood. An introduction to probabilistic programming, 2018.

Stefan Webb, Adam Golinski, Robert Zinkov, Siddharth Narayanaswamy, Tom Rainforth, Yee Whye Teh, and Frank Wood. Faithful inversion of generative models for effective amortized inference. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 3074–3084, 2018.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A comprehensive survey on graph neural networks. *CoRR*, abs/1901.00596, 2019.