Variational Inference with Normalizing Flows

Danilo Jimenez Rezende and Shakir Mohamed in ICML 2015

Presenter: Guangda Ji

21-951-249

guanji@student.ethz.ch

Overview

1. Framework of Variational Inference
   - Evidence Lower Bound
2. Previous Variational Inference Methods
   - Mean Field
   - Variational Auto Encoder and Deep Latent Gaussian Models
     - Decoder and Encoder
3. Design of Normalizing Flows
   - Definition
   - Choice of Flows
4. Experiments
   - Density Estimation
   - VI on Real Dataset
5. Discussion

Framework of Variational Inference

- Goal: design a model with parameter θ that asigns data $X = \{\mathbf{x}_i\}_{i=1}^N$ with high log-likelihood,

$$\text{Object 1: } \max_{\theta} \log p_\theta(X).$$

- Variational inference (VI) assume the model is generative.
  - Each $\mathbf{x}_i$ is determined by *latent variable* $\mathbf{z}$, $\mathbf{x} \sim p_\theta(\mathbf{x}|\mathbf{z})$. $\mathbf{z}$ has a prior of $p(\mathbf{z})$.
  - θ *generative parameter*, $p_\theta(\mathbf{x}|\mathbf{z})$ *Decoder*.
  - However, the margin $p_\theta(X) = \prod_{i=1}^N \int p(\mathbf{z}_i) p_\theta(\mathbf{x}_i|\mathbf{z}_i) \mathrm{d}\mathbf{z}_i$ is usually intractable.
- Therefore, we approximate the posterior with model $p(\mathbf{z}|\mathbf{x}) \approx q_\phi(\mathbf{z}|\mathbf{x})$.
  - ϕ *recognition parameter*
  - Achieve this by minimizing Kullback-Leibler divergence,

$$\text{Object 2: } \min_{\phi} \sum_{i=1}^N \mathsf{KL}(q_\phi(\cdot|\mathbf{x}_i) \| p_\theta(\cdot|\mathbf{x}_i)).$$

Evidence Lower Bound

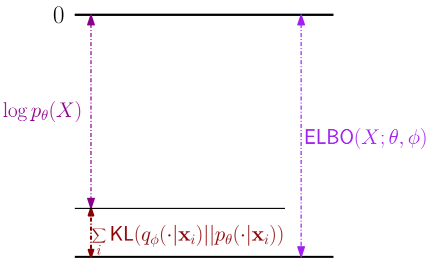
- We can rewrite the KL divergence as,

$$\mathsf{KL}(q_\phi(\cdot|\mathbf{x})\|p_\theta(\cdot|\mathbf{x})) = \mathop{\mathbb{E}}_{\mathbf{z}\sim q_\phi} \log\left(\frac{q_\phi(\mathbf{z}|\mathbf{x})}{p_\theta(\mathbf{z}|\mathbf{x})}\cdot\frac{p_\theta(\mathbf{x})}{p_\theta(\mathbf{x})}\right) = \mathop{\mathbb{E}}_{\mathbf{z}\sim q_\phi} \log\frac{q_\phi(\mathbf{z}|\mathbf{x})}{p(\mathbf{x},\mathbf{z})} + \log p_\theta(\mathbf{x})$$

$$= -\mathsf{ELBO}(\mathbf{x};\theta,\phi) + \log p_\theta(\mathbf{x}),$$

where $\mathsf{ELBO}(\mathbf{x};\theta,\phi) := \mathop{\mathbb{E}}_{\mathbf{z}\sim q_\phi}\left[\log\frac{p(\mathbf{x},\mathbf{z})}{q_\phi(\mathbf{z}|\mathbf{x})}\right] = -\mathsf{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p_\theta(\mathbf{x}|\mathbf{z})) + \mathop{\mathbb{E}}_{\mathbf{z}\sim q_\phi}\log p(\mathbf{z}).$

- Therefore, $\log p_\theta(X) = \sum_i \mathsf{ELBO}(\mathbf{x}_i;\theta,\phi) + \sum_i \mathsf{KL}(q_\phi(\cdot|\mathbf{x}_i)\|p_\theta(\cdot|\mathbf{x}_i))$
  - $\min_\phi \sum_i \mathsf{KL}(q_\phi(\cdot|\mathbf{x}_i)\|p_\theta(\cdot|\mathbf{x}_i)) \Leftrightarrow \max_\phi \sum_i \mathsf{ELBO}(\mathbf{x}_i;\theta,\phi).$
- $\mathsf{KL}\geq 0 \Rightarrow \log p_\theta(X) \geq \mathsf{ELBO}(X;\theta,\phi)$. $\mathsf{ELBO}(X;\theta,\phi)$ is the **E**vidence's **L**ower **BO**und.
  - Therefore, ELBO is a perfect objective function, $\max_{\phi,\theta}\mathsf{ELBO}(X;\theta,\phi)$.
  - This is called *Amortized Variational Inference* in the paper.



Then the challenges in Variational Inference have two aspects:

- How to design the class of posterior approximation function $q_\phi(\mathbf{z}|\mathbf{x})$ with enough richness,
- at the same time, efficient to optimize.

(naive) Mean Field approach Bishop [2006] (Don't need to ~~fully~~ understand)

- Assumption: split latent variable into sets
  - $\mathbf{z} = \mathbf{z}^{(1)} \times \mathbf{z}^{(2)} \times \cdots \times \mathbf{z}^{(M)}$

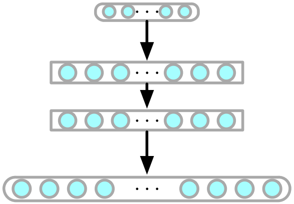
  $q_\phi(\mathbf{z}) = \prod_{i=1}^{M} q_i\left(\mathbf{z}_i\right)$, each $q_i(\cdot)$ is normalized.
- Fix other $\mathbf{z}_{-i} = \mathbf{z}/\mathbf{z}_i$, and optimize q_i
- A match of moments for some sufficient statistics, like EM algorithm.
- Disadvantages:
  1. Only feasible when $p(\mathbf{x}|\mathbf{z})$ is simple, and have analytical solution.
  2. $q(\mathbf{z})$ is unable to resemble the true posterior distribution.

Variational Auto Encoder and Deep Latent Gaussian Models: Generative Models (**Decoder**)

- Generative model is $p_\theta(x|\mathbf{z})$.
- VAE: $\mathcal{N}(\mathbf{x}|\mu_{\text{nn}}(\mathbf{z}), \sigma^2_{\text{nn}}(\mathbf{z}))$ for continuous $\mathbf{x}$, and $\mathrm{Ber}(\mathbf{x}|p_{\text{nn}}(\mathbf{z}), k_{\text{nn}}(\mathbf{z}))$ for binary $\mathbf{x}$.
- DLGM: the probability graph is $\mathbf{z}_L \to \mathbf{z}_{L-1} \to \cdots \to \mathbf{z}_1 \to \mathbf{x}$.
  - Each arrow $p(\cdot|\mathbf{z}_l)$ is a normal distribution with mean and variance comming from neural networks.

$$p\left(\mathbf{x}, \mathbf{z}_1, \ldots, \mathbf{z}_L\right) = p\left(\mathbf{x}|f_0\left(\mathbf{z}_1\right)\right) \prod_{l=1}^{L} p\left(\mathbf{z}_l|f_{l,\text{nn}}\left(\mathbf{z}_{l+1}\right)\right)$$

- Both models assume $p(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I})$.



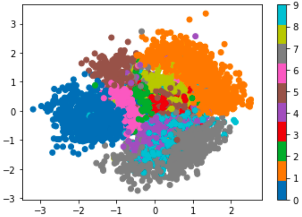
VAE and DLGM: Posterior Approximation (**Encoder**)

$$\mathsf{ELBO}(X;\theta,\phi) = -\mathsf{KL}(q_\phi(\mathbf{z}|X)\|p_\theta(X|\mathbf{z})) + \mathbb{E}_{q_\phi}\left[\log p(\mathbf{z})\right]$$

- Model for $q_\phi(Z|X)$: Gaussian, with parameters from a neural network

$$q_\phi(Z|X) = \prod_{i=1}^{N} \mathcal{N}(\mathbf{z}_i|\mu_{\mathsf{nn}}(\mathbf{x}_i), \sigma_{\mathsf{nn}}(\mathbf{x}_i)).$$

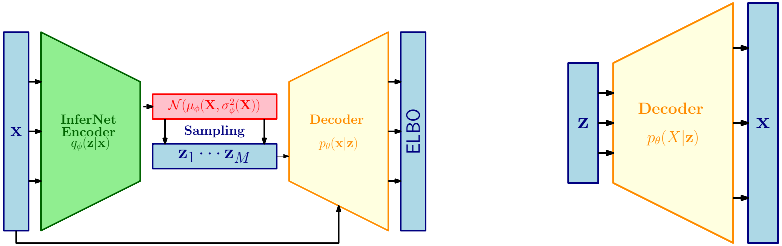
  - Named *inference network*.
  - It learns an inverse map from observations to latent variables.
  - Global parameter ϕ instead of computing $\mathbf{z}_i$ per data point.
  - Prior term $\underset{\mathbf{z}\sim q_\phi}{\mathbb{E}} \log p(\mathbf{z})$ in ELBO is explicit under Gaussian.



Projected latent space $\mathcal{Z}$ for MNIST.

- To maximize over ϕ, θ, we need
  1. $\nabla_\theta \mathsf{ELBO} = \mathbb{E}_{q_\phi}\left[\nabla_\theta \log p_\theta(X|\mathbf{z})\right], \nabla_\phi \mathsf{ELBO} = \nabla_\phi \mathbb{E}_{q_\phi}\left[\log p_\theta(X|\mathbf{z})\right] + \nabla_\phi \mathbb{E}_{q_\phi}\left[\log p(\mathbf{z})\right].$
  2. Intractable to integrate $\Rightarrow$ approximate with sampling, $\mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_i)}\left[f(\mathbf{z})\right] \approx \frac{1}{M}\sum_{m=1}^{M} f(\mathbf{z}^{(m)})$, where $\mathbf{z}^{(m)} \sim q_\phi(\cdot|\mathbf{x}_i)$.

Summary: $\mathrm{ELBO}(\mathbf{x};\theta,\phi) = -\mathrm{KL}(q_\phi(\mathbf{z}|\mathbf{x}) \| p_\theta(\mathbf{x}|\mathbf{z})) + \mathbb{E}_{q_\phi}[\log p(\mathbf{z})]$

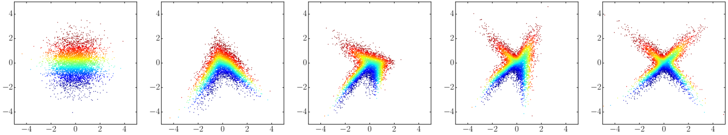


Training (left) and generating (right) diagram for VAE.

- CRITICS:
  - Only works on small datasets
  - Gaussian is too simple for approximating posterior
- Design new $q_\phi(\mathbf{z}|X)$ that
  1. density is explicit,
  2. efficient to sample from,
  3. have enough richness and complexity.

Definition of Normalizing Flows

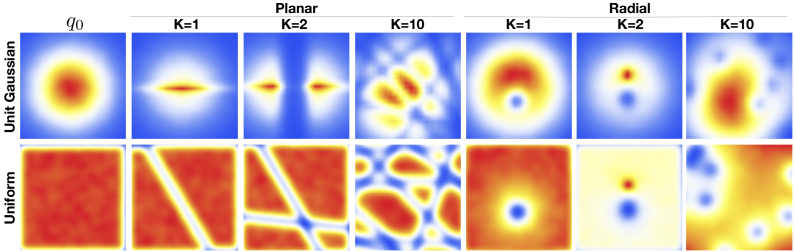
- Key Idea: $\mathbf{z} = f_K \circ \cdots \circ f_1(\mathbf{z}_0)$. Each f_k is simple and invertible. $\mathbf{z}_0$ has fixed, simple distribution, e.g., $q_0(\mathbf{z}_0) = \mathcal{N}(\mathbf{z}_0|0, \mathbf{I})$.
- Density is explicitly transformed by chain rule, $q(\mathbf{z}_k) = q(\mathbf{z}_{k-1}) \left| \det \frac{\partial f_k}{\partial_{\mathbf{z}_{k-1}}} \right|$
  - $\log q_K(\mathbf{z}_K) = \log q_0(\mathbf{z}_0) - \sum_{k=1}^{K} \log \left| \det \frac{\partial f_k}{\partial \mathbf{z}_{k-1}} \right|$
  - Efficient to sample, $\mathbb{E}_{q_K}[h(\mathbf{z})] = \mathbb{E}_{q_0}[h(f_K \circ f_{K-1} \circ \ldots \circ f_1(\mathbf{z}_0))]$.
  - Function composition gives large complexity.
  - Each f_k have parameters ϕ_k, $\phi = \{\phi_k\}_{k=1}^{K}$ is the total parameters.



A 4-step flow transforming samples from a standard-normal base density to a cross-shaped target density.

Flows in Practice: Finite Flows

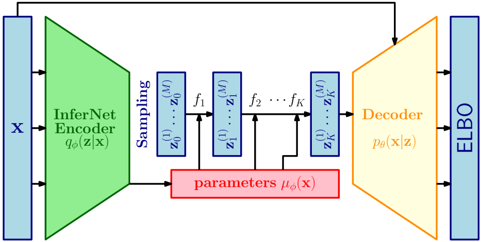
- Planar Flows: $f(\mathbf{z}) = \mathbf{z} + \mathbf{u}h\left(\mathbf{w}^\top \mathbf{z} + b\right)$, $h(\cdot)$ is $\tanh$ in this paper.
  - $\left|\det \frac{\partial f}{\partial \mathbf{z}}\right| = \left|\det\left(\mathbf{I} + \mathbf{u}\psi(\mathbf{z})^\top\right)\right| = \left|1 + \mathbf{u}^\top \psi(\mathbf{z})\right|$, where $\psi(\mathbf{z}) = h'\left(\mathbf{w}^\top \mathbf{z} + b\right)\mathbf{w}$
- Radial Flows: $f(\mathbf{z}) = \mathbf{z} + \beta h(\alpha, r)(\mathbf{z} - \mathbf{z}_0)$, where $r = |\mathbf{z} - \mathbf{z}_0|$ and $h(\alpha, r) = \frac{1}{\alpha + r}$.
  - $\left|\det \frac{\partial f}{\partial \mathbf{z}}\right| = [1 + \beta h(\alpha, r)]^{d-1}\left[1 + \beta h(\alpha, r) + \beta h'(\alpha, r) r\right)\right]$

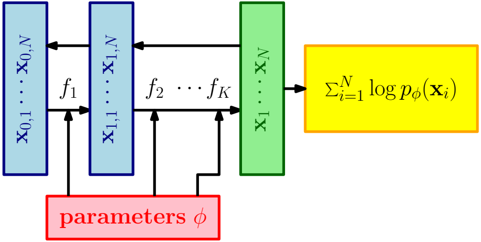


Diagrams for VI and density estimation

NF also be used for density estimation,

$$\min_{\phi} \mathrm{KL}\left(q_\phi(X) \| p(X)\right) \approx -\frac{1}{N}\sum_{n=1}^{N} \log q_\phi\left(\mathbf{x}_n\right) = -\frac{1}{N}\sum_{n=1}^{N} \log q_0\left(T_\phi^{-1}\left(\mathbf{x}_n\right)\right) + \log\left|J_{T_\phi^{-1}}\left(\mathbf{x}_n\right)\right| + \text{const.}$$





Training Diagram for variational inference (left) and density estimation (right) using normalizing flows.

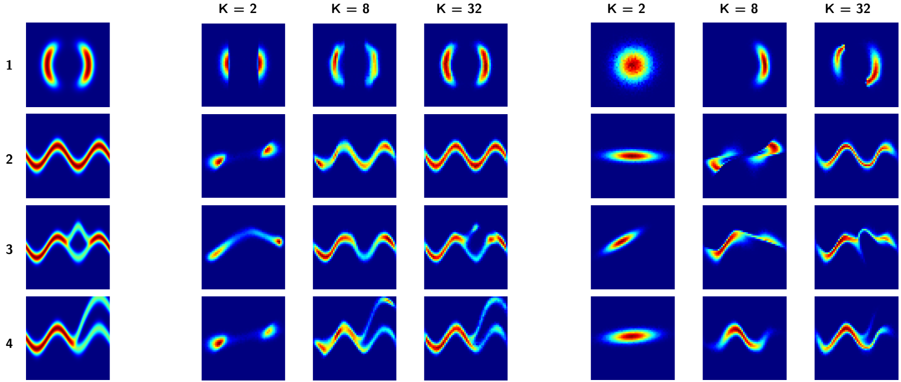
Flows in Practice: Non-linear Independent Components Estimation (NICE, Dinh et al. [2014])

- Key idea: Make each transformation volume preserved, $\left|\det \frac{\partial f_k}{\partial_{\mathbf{z}_k - 1}}\right| \equiv 1$
- Split $\mathbf{z} = (\mathbf{z}_A, \mathbf{z}_B) \in \mathbb{R}^d \times \mathbb{R}^{D-d}$, and transform as: $$f(\mathbf{z}) = (\mathbf{z}_A, \mathbf{z}_B + h_{\text{nn}}(\mathbf{z}_A))$$ $$f^{-1}(\mathbf{z}') = (\mathbf{z}'_A, \mathbf{z}'_B - h_{\text{nn}}(\mathbf{z}'_A))$$
- The Jacobian maxtrix is upper triangular: $\frac{\partial f}{\partial \mathbf{z}} = \begin{pmatrix} \mathbf{I}_d & \frac{\partial h_{\text{nn}}}{\partial \mathbf{z}_B} \\ 0 & \mathbf{I}_{D-d} \end{pmatrix}$
- Besides coupling, permutation and orthogonal transformation are used in order to add complexity.
- Mostly used flow. Also named *coupling flow*.

Comparison model: Hamiltonian variational approximation (HVI, Salimans et al. [2015])

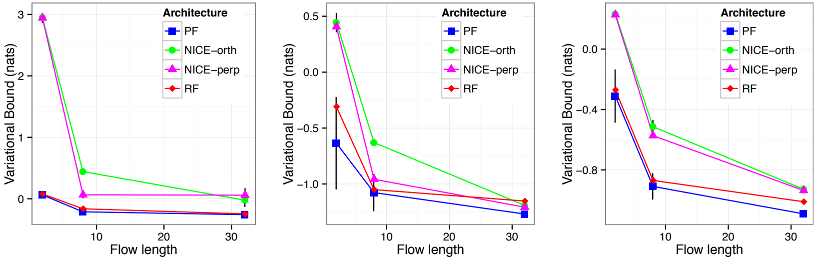
- Key idea: Hamiltonian Monte Carlo sampling for $p_\theta(\mathbf{z}|\mathbf{x})$
- Power: given any unnomrlized distribution $p_\theta(\mathbf{z}, \mathbf{x})$, the samples follow the normalized distribution.
- Two process, $\mathcal{T}_A \to \mathcal{T}_B \to \cdots \to \mathcal{T}_A \to \mathcal{T}_B$
  1. $\mathcal{T}_A$: Simulation for the motion of a partical in potential $U(\mathbf{z}) = -\log p_\theta(\mathbf{z}, \mathbf{x})$
  2. $\mathcal{T}_B$: Randomly resample the velocity of partical.
- Converge to Boltzmann distribution $p(\mathbf{z})p(\mathbf{v}) \to p_\theta(\mathbf{z}|\mathbf{x}) \times \mathcal{N}(\mathbf{v}|0, D\sigma^2)$
- Advantage is in sampling efficiency.

Experiments Results: Density Estimation, visual results



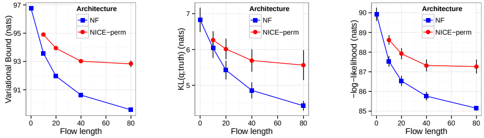
(a) True (b) Norm. Flow (c) NICE

Experiments Results: Density Estimation, ELBO results



Experiments Results: VI on Real Dataset, MNIST and CIFAR-10

Results on MNIST dataset	smaller better
Model	$-\log p(X)$
DLGM diagonal covariance	≤ 89.9
DLGM+NF (k = 10)	≤ 87.5
DLGM+NF (k = 20)	≤ 86.5
DLGM+NF (k = 40)	≤ 85.7
DLGM+NF (k = 80)	≤ 85.1
DLGM+NICE (k = 10)	≤ 88.6
DLGM+NICE (k = 20)	≤ 87.9
DLGM+NICE (k = 40)	≤ 87.3
DLGM+NICE (k = 80)	≤ 87.2
Results below from [Salimans et al., 2015]	
DLGM + HVI (1 leapfrog step)	88.08
DLGM + HVI (4 leapfrog steps)	86.40
DLGM + HVI (8 leapfrog steps)	85.51
Results below from [Gregor et al., 2014]	
DARN $n_h = 500$	84.71
DARN $n_h = 500$, adaNoise	84.13



(d) ELBO (e) KL$(q; p)$ (f) $-\log p(X)$

Figure: Effect of the flow-length on MNIST.

Table: Test set performance on the CIFAR-10 data.

	$K = 0$	$K = 2$	$K = 5$	$K = 10$
$-\log p(X)$	-293.7	-308.6	-317.9	-320.7

Discussion

- In stead of VI, density estimation is the major use case for normalizing flows.
- Planar and radial flows are no longer the predominat flows.
  1. The calculation for determinant is $O(D)$, for real application like images, dimension of $\mathbf{z}$ is usually high. This makes PF and RF inefficient.
  2. For density estimation task, $\max_\phi \sum_i \log q_K(\mathbf{y}\mathbf{x}_i)$, we also need to know $\mathcal{T}^{-1}(\mathbf{y}_i)$, finding inverse is not straighforward and need additional computation.
- Most flows follow the "NICE" fashion.

Autoregressive flows	Transformer type: – Affine – Combination-based – Integration-based – Spline-based	Conditioner type: – Recurrent – Masked – Coupling layer
Linear flows	Permutations Decomposition-based: – PLU – QR Orthogonal: – Exponential map – Cayley map – Householder	
Residual flows	Contractive residual Based on matrix determinant lemma: – Planar – Sylvester – Radial	

Table: Overview of methods for constructing flows based on finite compositions.

Thanks!

Flows in Practice: Non-linear Independent Components Estimation (NICE, Dinh et al. [2014])

- Can be more complex then a residual function:

$$\boldsymbol{y}^{(1:d)} = \boldsymbol{x}^{(1:d)}$$
$$\boldsymbol{y}^{(d+1:D)} = h\left(\boldsymbol{x}^{(d+1:D)}; f_{\theta,\text{nn}}\left(\boldsymbol{x}^{(1:d)}\right)\right)$$

- A lot of (major) contex on normalizing flows is about the design of this h abd f.

Infinitesimal Flows: Langevin Flow

- Considering $\lim_{K\to\infty} \mathbf{z} = f_K \circ \cdots \circ f_1(\mathbf{z}_0)$ and each transformation is infinitesimal $f_k = \mathbf{z}_k + d\mathbf{z}$
- Then the mapping $\mathbf{z}_0 \to \mathbf{z}_T$ turns into a differential equation.
- Consider the Stochastic process
  - Microscopic view, Langevin equation $d\mathbf{z}(t) = \mathbf{F}(\mathbf{z}(t), t)dt + \mathbf{G}(\mathbf{z}(t), t)d\boldsymbol{\xi}(t)$ for sampling.
  - Macroscopic view, Fokker-Planck equation
    $$\frac{\partial}{\partial t} q_t(\mathbf{z}) = -\sum_i \frac{\partial}{\partial z_i}\left[F_i(\mathbf{z}, t) q_t\right] + \frac{1}{2}\sum_{i,j}\frac{\partial^2}{\partial z_i \partial z_j}\left[D_{ij}(\mathbf{z}, t) q_t\right]$$
  - The asymptotic distribution is explicity, $q_\infty \propto \exp\left(-\int F \mathrm{d}\mathbf{z}\right)$.
  - If we set $F(\mathbf{z}, t) = -\nabla_z \log p(\mathbf{z}, X)$, then we will converge to $q_\infty = p_\theta(\mathbf{z}|X)$.

Further Readings for this paper Rezende and Mohamed [2015]

- Bishop [2006] Chapter 10 provide detailed calculation using mean field methods.
- Kingma and Welling [2013] and Rezende et al. [2014] describe the framework of variational autoencoder.
- Papamakarios et al. [2021] and Kobyzev et al. [2020] are very good reviews on normalizing flows.
- Ganguly and Earp [2021] provides very basic and solid knowledge on variational inference.
- Bond-Taylor et al. [2021] is a very comprehensive review on all deep generative methods, including GAN, VAE, NF, energy based and autoregressive methods.
- Salimans et al. [2015] shows in detail how to combine VI with HMC
- Neal et al. [2011] provides basic knowledge on MCMC with Hamiltonians (HMC).
- Welling and Teh [2011] provides an example in machine learning that uses Langevin dynamics in sampling.

References I

C. M. Bishop. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg, 2006. ISBN 0387310738.

S. Bond-Taylor, A. Leach, Y. Long, and C. G. Willcocks. Deep generative modelling: A comparative review of vaes, gans, normalizing flows, energy-based and autoregressive models. arXiv preprint arXiv:2103.04922, 2021.

L. Dinh, D. Krueger, and Y. Bengio. Nice: Non-linear independent components estimation. arXiv preprint arXiv:1410.8516, 2014.

A. Ganguly and S. W. Earp. An introduction to variational inference. arXiv preprint arXiv:2108.13083, 2021.

K. Gregor, I. Danihelka, A. Mnih, C. Blundell, and D. Wierstra. Deep autoregressive networks. In International Conference on Machine Learning, pages 1242–1250. PMLR, 2014.

D. P. Kingma and M. Welling. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114, 2013.

References II

I. Kobyzev, S. J. Prince, and M. A. Brubaker. Normalizing flows: An introduction and review of current methods. IEEE transactions on pattern analysis and machine intelligence, 43(11):3964–3979, 2020.

R. M. Neal et al. Mcmc using hamiltonian dynamics. Handbook of markov chain monte carlo, 2(11):2, 2011.

G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. Journal of Machine Learning Research, 22(57):1–64, 2021.

D. Rezende and S. Mohamed. Variational inference with normalizing flows. In International conference on machine learning, pages 1530–1538. PMLR, 2015.

D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In International conference on machine learning, pages 1278–1286. PMLR, 2014.

References III

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

M. Welling and Y. W. Teh. Bayesian learning via stochastic gradient langevin dynamics. In Proceedings of the 28th international conference on machine learning (ICML-11), pages 681–688. Citeseer, 2011.