Score function estimator and variance reduction techniques

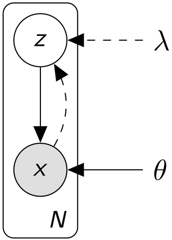
Wilker Aziz
University of Amsterdam

May 16, 2018

Outline

Variational inference for belief networks

Generative model with NN likelihood



Let $z \in \{0, 1\}^d$ and

$$\begin{aligned} q_\lambda(z|x) &= \prod_{i=1}^{d} q_\lambda(z_i|x) \\ &= \prod_{i=1}^{d} \mathrm{Bern}(z_i|\,\mathrm{sigmoid}(f_\lambda(x))) \end{aligned} \tag{1}$$

Jointly optimise generative model $p_\theta(x|z)$ and inference model $q_\lambda(z|x)$ under the same objective (ELBO)

Mnih and Gregor (2014)

Objective

$$\log p_\theta(x) \geq \overbrace{\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x, Z)\right] + \mathbb{H}\left(q_\lambda(z|x)\right)}^{\text{ELBO}}$$
$$= \mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|Z)\right] - \mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)$$

Parameter estimation

$$\arg\max_{\theta,\lambda} \; \mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|Z)\right] - \mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)$$

Generative Network Gradient

$$\frac{\partial}{\partial\theta}\left(\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] - \overbrace{\mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)}^{\text{constant wrt } \theta}\right)$$

Generative Network Gradient

$$\frac{\partial}{\partial \theta}\left(\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] - \overbrace{\mathrm{KL}\left(q_\lambda(z|x)\ ||\ p(z)\right)}^{\text{constant wrt } \theta}\right)$$

$$= \underbrace{\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\partial}{\partial \theta}\log p_\theta(x|z)\right]}_{\text{expected gradient :)}}$$

Generative Network Gradient

$$\frac{\partial}{\partial \theta}\left(\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] - \overbrace{\mathrm{KL}\left(q_\lambda(z|x) \,||\, p(z)\right)}^{\text{constant wrt } \theta}\right)$$

$$= \underbrace{\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\partial}{\partial \theta}\log p_\theta(x|z)\right]}_{\text{expected gradient :)}}$$

$$\overset{\text{MC}}{\approx} \frac{1}{K}\sum_{k=1}^{K}\frac{\partial}{\partial \theta}\log p_\theta(x|z^{(k)})$$

$$z^{(k)} \sim q_\lambda(Z|x)$$

Generative Network Gradient

$$\frac{\partial}{\partial \theta}\left(\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] - \overbrace{\mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)}^{\text{constant wrt } \theta}\right)$$

$$= \underbrace{\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\partial}{\partial \theta}\log p_\theta(x|z)\right]}_{\text{expected gradient :)}}$$

$$\overset{\text{MC}}{\approx} \frac{1}{K}\sum_{k=1}^{K}\frac{\partial}{\partial \theta}\log p_\theta(x|z^{(k)})$$

$$z^{(k)} \sim q_\lambda(Z|x)$$

Note: $q_\lambda(z|x)$ does not depend on θ.

Inference Network Gradient

$$\frac{\partial}{\partial \lambda} \left(\mathbb{E}_{q_\lambda(z|x)} \left[\log p_\theta(x|z) \right] - \overbrace{\mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z) \right)}^{\text{analytical}} \right)$$

Inference Network Gradient

$$\frac{\partial}{\partial \lambda}\left(\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] - \overbrace{\mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)}^{\text{analytical}}\right)$$

$$= \frac{\partial}{\partial \lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] - \underbrace{\frac{\partial}{\partial \lambda}\mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)}_{\text{analytical computation}}$$

Inference Network Gradient

$$\frac{\partial}{\partial \lambda}\left(\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] - \overbrace{\mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)}^{\text{analytical}}\right)$$

$$= \frac{\partial}{\partial \lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] - \underbrace{\frac{\partial}{\partial \lambda}\mathrm{KL}\left(q_\lambda(z|x) \mid\mid p(z)\right)}_{\text{analytical computation}}$$

The first term again requires approximation by sampling, but there is a problem

Inference Network Gradient

$$\frac{\partial}{\partial\lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right]$$

Inference Network Gradient

$$\frac{\partial}{\partial \lambda} \mathbb{E}_{q_\lambda(z|x)} [\log p_\theta(x|z)]$$
$$= \frac{\partial}{\partial \lambda} \int q_\lambda(z|x) \log p_\theta(x|z) \mathrm{d}z$$

Inference Network Gradient

$$
\begin{aligned}
&\frac{\partial}{\partial \lambda} \mathbb{E}_{q_\lambda(z|x)} [\log p_\theta(x|z)] \\
&= \frac{\partial}{\partial \lambda} \int q_\lambda(z|x) \log p_\theta(x|z) \mathrm{d}z \\
&= \underbrace{\int {\color{red}\frac{\partial}{\partial \lambda}(q_\lambda(z|x))} \log p_\theta(x|z) \, \mathrm{d}z}_{\text{not an expectation}}
\end{aligned}
$$

Inference Network Gradient

$$
\begin{aligned}
&\frac{\partial}{\partial \lambda} \mathbb{E}_{q_\lambda(z|x)} [\log p_\theta(x|z)] \\
&= \frac{\partial}{\partial \lambda} \int q_\lambda(z|x) \log p_\theta(x|z) \mathrm{d}z \\
&= \underbrace{\int \frac{\partial}{\partial \lambda}(q_\lambda(z|x)) \log p_\theta(x|z)\, \mathrm{d}z}_{\text{not an expectation}}
\end{aligned}
$$

- MC estimator is non-differentiable: cannot sample first

Inference Network Gradient

$$
\begin{aligned}
&\frac{\partial}{\partial \lambda} \mathbb{E}_{q_\lambda(z|x)} [\log p_\theta(x|z)] \\
&= \frac{\partial}{\partial \lambda} \int q_\lambda(z|x) \log p_\theta(x|z) \mathrm{d}z \\
&= \underbrace{\int \frac{\partial}{\partial \lambda}(q_\lambda(z|x)) \log p_\theta(x|z) \, \mathrm{d}z}_{\text{not an expectation}}
\end{aligned}
$$

- MC estimator is non-differentiable: cannot sample first
- Differentiating the expression does not yield an expectation: cannot approximate via MC

Outline

Score function estimator

We can again use the log identity for derivatives

$$
\begin{aligned}
&\frac{\partial}{\partial\lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] \\
&= \frac{\partial}{\partial\lambda}\int q_\lambda(z|x)\log p_\theta(x|z)\mathrm{d}z \\
&= \underbrace{\int \frac{\partial}{\partial\lambda}(q_\lambda(z|x))\log p_\theta(x|z)\,\mathrm{d}z}_{\text{not an expectation}}
\end{aligned}
$$

Score function estimator

We can again use the log identity for derivatives

$$
\begin{aligned}
&\frac{\partial}{\partial \lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] \\
&= \frac{\partial}{\partial \lambda}\int q_\lambda(z|x)\log p_\theta(x|z)\mathrm{d}z \\
&= \underbrace{\int \frac{\partial}{\partial \lambda}(q_\lambda(z|x))\log p_\theta(x|z)\,\mathrm{d}z}_{\text{not an expectation}} \\
&= \int q_\lambda(z|x)\frac{\partial}{\partial \lambda}(\log q_\lambda(z|x))\log p_\theta(x|z)\,\mathrm{d}z
\end{aligned}
$$

Score function estimator

We can again use the log identity for derivatives

$$
\begin{aligned}
&\frac{\partial}{\partial \lambda} \mathbb{E}_{q_\lambda(z|x)} [\log p_\theta(x|z)] \\
&= \frac{\partial}{\partial \lambda} \int q_\lambda(z|x) \log p_\theta(x|z) \mathrm{d}z \\
&= \underbrace{\int \frac{\partial}{\partial \lambda} (q_\lambda(z|x)) \log p_\theta(x|z) \, \mathrm{d}z}_{\text{not an expectation}} \\
&= \int q_\lambda(z|x) \frac{\partial}{\partial \lambda} (\log q_\lambda(z|x)) \log p_\theta(x|z) \, \mathrm{d}z \\
&= \underbrace{\mathbb{E}_{q_\lambda(z|x)} \left[\log p_\theta(x|z) \frac{\partial}{\partial \lambda} \log q_\lambda(z|x) \right]}_{\text{expected gradient :)}}
\end{aligned}
$$

Score function estimator: high variance

We can now build an MC estimator

$$\frac{\partial}{\partial \lambda} \mathbb{E}_{q_\lambda(z|x)} [\log p_\theta(x|z)]$$

$$= \mathbb{E}_{q_\lambda(z|x)} \left[\log p_\theta(x|z) \frac{\partial}{\partial \lambda} \log q_\lambda(z|x) \right]$$

Score function estimator: high variance

We can now build an MC estimator

$$\frac{\partial}{\partial \lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right]$$

$$= \mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\frac{\partial}{\partial \lambda}\log q_\lambda(z|x)\right]$$

$$\overset{\text{MC}}{\approx} \frac{1}{K}\sum_{k=1}^{K} \log p_\theta(x|z^{(k)})\frac{\partial}{\partial \lambda}\log q_\lambda(z^{(k)}|x)$$

$$z^{(k)} \sim q_\lambda(Z|x)$$

Score function estimator: high variance

We can now build an MC estimator

$$
\begin{aligned}
&\frac{\partial}{\partial \lambda} \mathbb{E}_{q_\lambda(z|x)} [\log p_\theta(x|z)] \\
&= \mathbb{E}_{q_\lambda(z|x)} \left[\log p_\theta(x|z) \frac{\partial}{\partial \lambda} \log q_\lambda(z|x) \right] \\
&\overset{\text{MC}}{\approx} \frac{1}{K} \sum_{k=1}^{K} \color{red}{\log p_\theta(x|z^{(k)})} \frac{\partial}{\partial \lambda} \log q_\lambda(z^{(k)}|x) \\
&z^{(k)} \sim q_\lambda(Z|x)
\end{aligned}
$$

but

- magnitude of $\log p_\theta(x|z)$ varies widely

Score function estimator: high variance

We can now build an MC estimator

$$
\begin{aligned}
&\frac{\partial}{\partial \lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] \\
&= \mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\frac{\partial}{\partial \lambda}\log q_\lambda(z|x)\right] \\
&\overset{\text{MC}}{\approx} \frac{1}{K}\sum_{k=1}^{K} {\color{red}\log p_\theta(x|z^{(k)})}\frac{\partial}{\partial \lambda}\log q_\lambda(z^{(k)}|x) \\
&z^{(k)} \sim q_\lambda(Z|x)
\end{aligned}
$$

but

- magnitude of $\log p_\theta(x|z)$ varies widely
- model likelihood does not contribute to direction of gradient

Score function estimator: high variance

We can now build an MC estimator

$$\begin{aligned}
&\frac{\partial}{\partial\lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] \\
&= \mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] \\
&\overset{\text{MC}}{\approx} \frac{1}{K}\sum_{k=1}^{K} {\color{red}\log p_\theta(x|z^{(k)})}\frac{\partial}{\partial\lambda}\log q_\lambda(z^{(k)}|x) \\
&z^{(k)} \sim q_\lambda(Z|x)
\end{aligned}$$

but

- magnitude of $\log p_\theta(x|z)$ varies widely
- model likelihood does not contribute to direction of gradient
- too much variance to be useful

Score function estimator: high variance

We can now build an MC estimator

$$
\begin{aligned}
&\frac{\partial}{\partial \lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] \\
&= \mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\frac{\partial}{\partial \lambda}\log q_\lambda(z|x)\right] \\
&\overset{\text{MC}}{\approx} \frac{1}{K}\sum_{k=1}^{K} {\color{red}\log p_\theta(x|z^{(k)})}\frac{\partial}{\partial \lambda}\log q_\lambda(z^{(k)}|x) \\
&z^{(k)} \sim q_\lambda(Z|x)
\end{aligned}
$$

but

- magnitude of $\log p_\theta(x|z)$ varies widely
- model likelihood does not contribute to direction of gradient
- too much variance to be useful

but

Score function estimator: high variance

We can now build an MC estimator

$$\begin{aligned}
&\frac{\partial}{\partial \lambda}\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\right] \\
&= \mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\frac{\partial}{\partial \lambda}\log q_\lambda(z|x)\right] \\
&\overset{\text{MC}}{\approx} \frac{1}{K}\sum_{k=1}^{K} {\color{red}\log p_\theta(x|z^{(k)})}\frac{\partial}{\partial \lambda}\log q_\lambda(z^{(k)}|x) \\
&z^{(k)} \sim q_\lambda(Z|x)
\end{aligned}$$

but

- magnitude of $\log p_\theta(x|z)$ varies widely
- model likelihood does not contribute to direction of gradient
- too much variance to be useful

but **fully differentiable!**

When variance is high we can

- sample more

When variance is high we can

- sample more
  won't scale

When variance is high we can

- sample more
  won't scale
- use variance reduction techniques (e.g. baselines and control variates)

When variance is high we can

- sample more
  won't scale
- use variance reduction techniques (e.g. baselines and control variates)
  excellent idea!

When variance is high we can

- sample more
  won't scale
- use variance reduction techniques (e.g. baselines and control variates)
  excellent idea!
  and now it's time for it!

Example Model

Let us consider a latent factor model for topic modelling:

Example Model

Let us consider a latent factor model for topic modelling:

- a document $x = (x_1, \ldots, x_n)$ consists of n i.i.d. categorical draws from that model

Example Model

Let us consider a latent factor model for topic modelling:

- a document $x = (x_1, \ldots, x_n)$ consists of n i.i.d. categorical draws from that model
- the categorical distribution in turn depends on the binary latent factors $z = (z_1, \ldots, z_k)$ which are also i.i.d.

Example Model

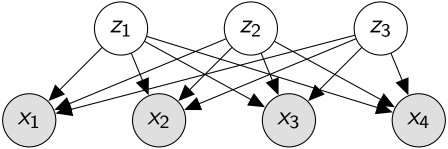
Let us consider a latent factor model for topic modelling:

- a document $x = (x_1, \ldots, x_n)$ consists of n i.i.d. categorical draws from that model
- the categorical distribution in turn depends on the binary latent factors $z = (z_1, \ldots, z_k)$ which are also i.i.d.

$$\begin{aligned} z_j &\sim \text{Bernoulli}\,(\phi) & (1 \leq j \leq k) \\ x_i &\sim \text{Categorical}\,(g_\theta(z)) & (1 \leq i \leq n) \end{aligned}$$

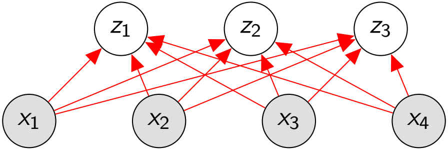
Here ϕ specifies a Bernoulli prior and $g_\theta(\cdot)$ is a function computed by neural network with softmax output.

Example Model



At inference time the latent variables are marginally dependent. For our variational distribution we are going to assume that they are not (recall: mean field assumption).

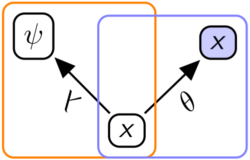
Inference Network



The inference network needs to predict k Bernoulli parameters ψ. Any neural network with sigmoid output will do that job.

$$q_\lambda(z|x) = \prod_{i=1}^{k} \text{Bern}(z_i|\psi_i) \quad \text{where } \psi = f_\lambda(x) \tag{2}$$

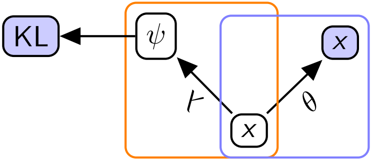
Computation Graph



inference model

generation model

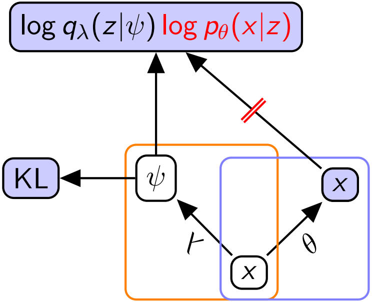
Computation Graph



inference model

generation model

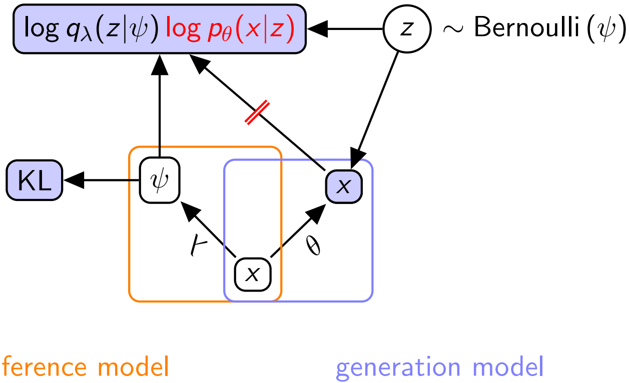
Computation Graph



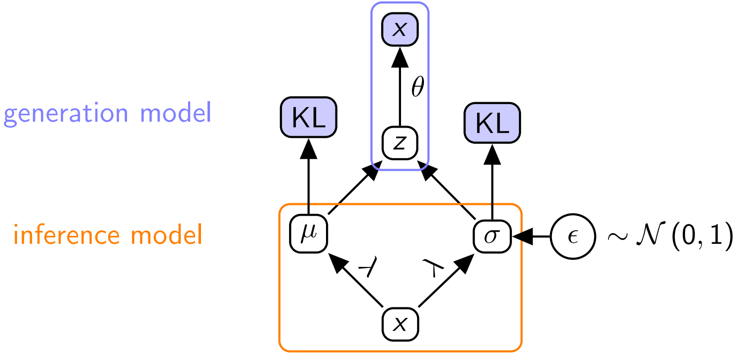
inference model

generation model

Computation Graph



Reparametrisation Gradient



Pros and Cons

- Pros
  - Applicable to all distributions
  - Many libraries come with samplers for common distributions

Pros and Cons

- Pros
  - Applicable to all distributions
  - Many libraries come with samplers for common distributions
- Cons
  - High Variance!

Outline

1. Recap
2. Score function estimator
3. Variance reduction

Baselines

$$
\begin{aligned}
&\mathbb{E}_{q_\lambda(z|x)}\left[\log p_\theta(x|z)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right]\\
&=\mathbb{E}_{q_\lambda(z|x)}\left[\left(\log p_\theta(x|z)-C(x)\right)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right]\\
&+\mathbb{E}_{q_\lambda(z|x)}\left[C(x)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right]\\
&=\mathbb{E}_{q_\lambda(z|x)}\left[\left(\log p_\theta(x|z)-C(x)\right)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right]+C(x)
\end{aligned}
\tag{3}
$$

Baselines

We attempt to centre the gradient estimate. To do this we learn a quantity C that we subtract from the reconstruction loss.

$$\log q(z|\lambda) \left(\log p(x|z, \theta) - C\right)$$

We call C a baseline. It does not change the expected gradient (Williams, 1992).

Baselines

We can make baselines input-depdendent to make them more flexible.

$$\log q(z|\lambda) \left(\log p(x|z, \theta) - C(x)\right)$$

However, baselines may not depend on the random value z!
Quantities that may depend on the random value ($C(z)$) are called **control variates**. See Blei et al. (2012); Ranganath et al. (2014); Gregor et al. (2014).

Baselines

Baselines are predicted by a regression model (e.g. a neural net).
The model is trained using an L_2-loss.

$$\min \left(C(x) - \log p(x|z, \theta) \right)^2$$

Summary

Summary

- Reparametrisation not available for discrete variables.

Summary

- Reparametrisation not available for discrete variables.
- Use score function estimator.

Summary

- Reparametrisation not available for discrete variables.
- Use score function estimator.
- High variance.

Summary

- Reparametrisation not available for discrete variables.
- Use score function estimator.
- High variance.
- Always use baselines for variance reduction!

Literature I

David M. Blei, Michael I. Jordan, and John W. Paisley. Variational bayesian inference with stochastic search. In *ICML*, 2012. URL http://icml.cc/2012/papers/687.pdf.

Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. Deep autoregressive networks. In Eric P. Xing and Tony Jebara, editors, *ICML*, pages 1242–1250, 2014. URL http://proceedings.mlr.press/v32/gregor14.html.

Andriy Mnih and Karol Gregor. Neural variational inference and learning in belief networks. *arXiv preprint arXiv:1402.0030*, 2014.

Rajesh Ranganath, Sean Gerrish, and David Blei. Black Box Variational Inference. In Samuel Kaski and Jukka Corander, editors, *AISTATS*, pages 814–822, 2014. URL http://proceedings.mlr.press/v33/ranganath14.pdf.

Literature II

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4): 229–256, 1992. URL https://doi.org/10.1007/BF00992696.