# Score function estimator and variance reduction techniques

Wilker Aziz
University of Amsterdam

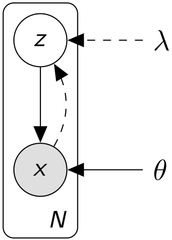
May 24, 2018

# Outline

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

# Variational inference for belief networks

Generative model with NN likelihood




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

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

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

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

Parameter estimation

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

# Generative Network Gradient

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

# Generative Network Gradient

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

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

# 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

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

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

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

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

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

# Inference Network Gradient

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

# 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

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

# Inference Network Gradient

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

# Inference Network Gradient

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

- MC estimator is non-differentiable: cannot sample first

# Inference Network Gradient

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

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

# Inference Network Gradient

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

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

The original VAE employed a *reparameterisation*, but...

# Bernoulli pmf

$$\mathrm{Bern}(z_j|b_j) = b_j^{z_j}(1 - b_j)^{1-z_j}$$

# Bernoulli pmf

$$
\begin{aligned}
\mathrm{Bern}(z_j|b_j) &= b_j^{z_j}(1-b_j)^{1-z_j} \\
&= \begin{cases} b_j & \text{if } z_j = 1 \\ 1-b_j & \text{if } z_j = 0 \end{cases}
\end{aligned}
$$

# Bernoulli pmf

$$\begin{aligned} \mathrm{Bern}(z_j|b_j) &= b_j^{z_j}(1-b_j)^{1-z_j} \\ &= \begin{cases} b_j & \text{if } z_j = 1 \\ 1-b_j & \text{if } z_j = 0 \end{cases} \end{aligned}$$

Can we reparameterise a Bernoulli variable?

# Reparameterisation requires a Jacobian matrix

Not really :(

$$q(z;\lambda) = \underbrace{\phi(\epsilon = h(z,\lambda))\left|\det J_{h(z,\lambda)}\right|}_{\text{change of density}} \tag{2}$$

Elements in the Jacobian matrix

$$J_{h(z,\lambda)}[i,j] = \frac{\partial h_i(z,\lambda)}{\partial z_j}$$

are not defined for non-differentiable functions

# 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

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

# Score function estimator: high variance

We can now build an MC estimator

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

# Score function estimator: high variance

We can now build an MC estimator

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

# 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

# Score function estimator: high variance

We can now build an MC estimator

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

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

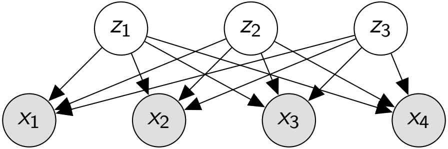
Let us consider a latent factor model for topic modelling:

- 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.

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

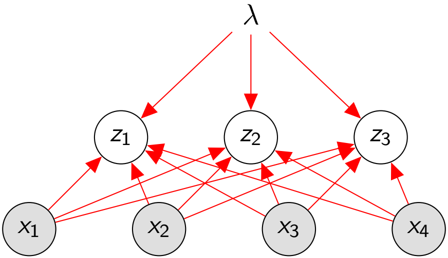
Here $\phi$ specifies a Bernoulli prior and $g_\theta(\cdot)$ is a function computed by neural network with softmax output.

# 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).

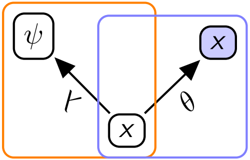
# Inference Network




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

$$q_\lambda(z|x) = \prod_{j=1}^{k} \text{Bern}(z_j|\psi_j) \quad \text{where } \psi_1^k = f_\lambda(x) \tag{3}$$

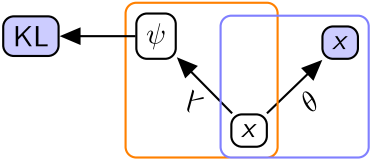
# Computation Graph




inference model

generation model

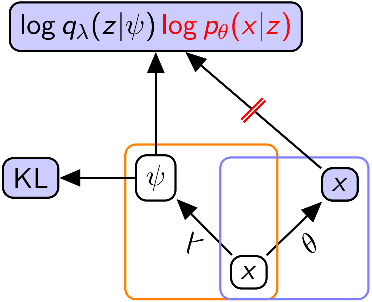
# Computation Graph




inference model

generation model

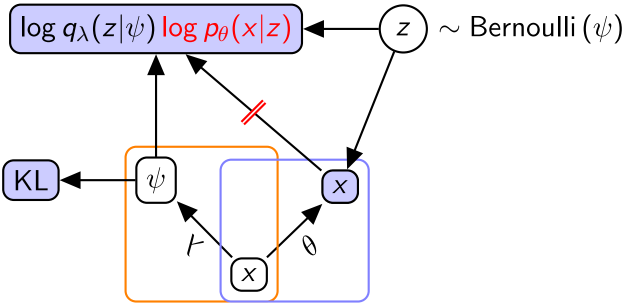
# Computation Graph




inference model

generation model

# Computation Graph



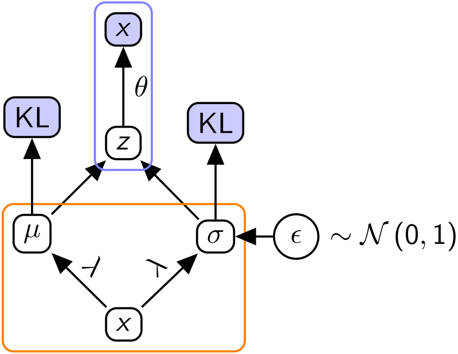

inference model

generation model

# Reparametrisation Gradient

generation model

inference model



# 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

# Control variates

Suppose we want to estimate $\mathbb{E}[f(Z)]$ and we know the expected value of another function $\psi(z)$ on the same support.

Greensmith et al. (2004)

# Control variates

Suppose we want to estimate $\mathbb{E}[f(Z)]$ and we know the expected value of another function $\psi(z)$ on the same support.

Then it holds that

$$\mathbb{E}[f(Z)] = \mathbb{E}[f(Z) - \psi(Z)] + \mathbb{E}[\psi(Z)] \tag{4}$$

Greensmith et al. (2004)

## Control variates

Suppose we want to estimate $\mathbb{E}[f(Z)]$ and we know the expected value of another function $\psi(z)$ on the same support.

Then it holds that

$$\mathbb{E}[f(Z)] = \mathbb{E}[f(Z) - \psi(Z)] + \mathbb{E}[\psi(Z)] \tag{4}$$

If $\psi(z) = f(z)$, and we estimate the expected value of $f(x) - \psi(x)$, then we have reduced variance to 0.

Greensmith et al. (2004)

## Control variates

Suppose we want to estimate $\mathbb{E}[f(Z)]$ and we know the expected value of another function $\psi(z)$ on the same support.

Then it holds that

$$\mathbb{E}[f(Z)] = \mathbb{E}[f(Z) - \psi(Z)] + \mathbb{E}[\psi(Z)] \tag{4}$$

If $\psi(z) = f(z)$, and we estimate the expected value of $f(x) - \psi(x)$, then we have reduced variance to 0. In general

$$\mathrm{Var}(f - \psi) = \mathrm{Var}(f) - 2\,\mathrm{Cov}(f, \psi) + \mathrm{Var}(\psi) \tag{5}$$

If $f$ and $\psi$ are strongly correlated and the covariance is greater than $\mathrm{Var}(\psi)$, then we improve on the original estimation problem.

Greensmith et al. (2004)

# Reducing variance of score function estimator

Back to the score function estimator

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

# Reducing variance of score function estimator

Back to the score function estimator

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

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

$$+ \mathbb{E}_{q_\lambda(z|x)}\left[\underbrace{C(x)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)}_{\psi(z)}\right]$$

The last term is very simple!

# $\mathbb{E}[\psi(Z)]$

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

# $\mathbb{E}[\psi(Z)]$

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

# $\mathbb{E}[\psi(Z)]$

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

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

# $\mathbb{E}[\psi(Z)]$

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

$$= C(x)\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{q_\lambda(z|x)}\right] = C(x)\int q_\lambda(z|x)\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{q_\lambda(z|x)}\,\mathrm{d}z$$

# $\mathbb{E}[\psi(Z)]$

$$\mathbb{E}_{q_\lambda(z|x)}\left[C(x)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] = C(x)\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right]$$
$$= C(x)\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{q_\lambda(z|x)}\right] = C(x)\int \color{red}{q_\lambda(z|x)}\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{\color{red}{q_\lambda(z|x)}}\,\mathrm{d}z$$
$$= C(x)\int \frac{\partial}{\partial\lambda}q_\lambda(z|x)\,\mathrm{d}z$$

# $\mathbb{E}[\psi(Z)]$

$$\begin{aligned}
\mathbb{E}_{q_\lambda(z|x)}\left[C(x)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] &= C(x)\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] \\
= C(x)\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{q_\lambda(z|x)}\right] &= C(x)\int q_\lambda(z|x)\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{q_\lambda(z|x)}\,\mathrm{d}z \\
= C(x)\int\frac{\partial}{\partial\lambda}q_\lambda(z|x)\,\mathrm{d}z &= C(x)\frac{\partial}{\partial\lambda}\int q_\lambda(z|x)\,\mathrm{d}z
\end{aligned}$$

# $\mathbb{E}[\psi(Z)]$

$$
\begin{aligned}
\mathbb{E}_{q_\lambda(z|x)}\left[C(x)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] &= C(x)\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] \\
= C(x)\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{q_\lambda(z|x)}\right] &= C(x)\int {\color{red}q_\lambda(z|x)}\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{{\color{red}q_\lambda(z|x)}}\,\mathrm{d}z \\
= C(x)\int\frac{\partial}{\partial\lambda}q_\lambda(z|x)\,\mathrm{d}z &= C(x)\frac{\partial}{\partial\lambda}\int q_\lambda(z|x)\,\mathrm{d}z \\
&= C(x)\frac{\partial}{\partial\lambda}1
\end{aligned}
$$

# $\mathbb{E}[\psi(Z)]$

$$\mathbb{E}_{q_\lambda(z|x)}\left[C(x)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] = C(x)\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right]$$
$$= C(x)\mathbb{E}_{q_\lambda(z|x)}\left[\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{q_\lambda(z|x)}\right] = C(x)\int {\color{red}q_\lambda(z|x)}\frac{\frac{\partial}{\partial\lambda}q_\lambda(z|x)}{{\color{red}q_\lambda(z|x)}}\,\mathrm{d}z$$
$$= C(x)\int\frac{\partial}{\partial\lambda}q_\lambda(z|x)\,\mathrm{d}z = C(x)\frac{\partial}{\partial\lambda}\int q_\lambda(z|x)\,\mathrm{d}z$$
$$= C(x)\frac{\partial}{\partial\lambda}1 = 0$$

# Improved estimator

Back to the score function estimator

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

# Improved estimator

Back to the score function estimator

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

# Improved estimator

Back to the score function estimator

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

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

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

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

# Improved estimator

Back to the score function estimator

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

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

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

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

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

$C(x)$ is called a **baseline**

# Baselines

Baselines can be constant

$$\mathbb{E}_{q_\lambda(z|x)}\left[\left(\log p_\theta(x|z) - C\right)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] \tag{6}$$

Williams (1992)

# Baselines

Baselines can be constant

$$\mathbb{E}_{q_\lambda(z|x)}\left[\left(\log p_\theta(x|z) - C\right)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] \tag{6}$$

or input-dependent

$$\mathbb{E}_{q_\lambda(z|x)}\left[\left(\log p_\theta(x|z) - C(x)\right)\frac{\partial}{\partial\lambda}\log q_\lambda(z|x)\right] \tag{7}$$

Williams (1992)

## Baselines

Baselines can be constant

$$\mathbb{E}_{q_\lambda(z|x)}\left[\left(\log p_\theta(x|z) - C\right) \frac{\partial}{\partial \lambda} \log q_\lambda(z|x)\right] \tag{6}$$

or input-dependent

$$\mathbb{E}_{q_\lambda(z|x)}\left[\left(\log p_\theta(x|z) - C(x)\right) \frac{\partial}{\partial \lambda} \log q_\lambda(z|x)\right] \tag{7}$$

or both

$$\mathbb{E}_{q_\lambda(z|x)}\left[\left(\log p_\theta(x|z) - C - C(x)\right) \frac{\partial}{\partial \lambda} \log q_\lambda(z|x)\right] \tag{8}$$

---

Williams (1992)

# Full power of control variates

If we design $C(\cdot)$ to depend on the variable of integration $z$, we exploit the full power of **control variates**, but designing and using those require more careful treatment

Blei et al. (2012); Ranganath et al. (2014); Gregor et al. (2014)

# Learning baselines

Baselines are predicted by a regression model (e.g. a neural net).

One idea is to "centre the learning signal", in which case we train the baseline with an $L_2$-loss:

$$\rho = \arg\min_{\rho} \left( C_\rho(x) - \log p(x|z) \right)^2$$

Gu et al. (2015)

# Putting it together

Parameter estimation

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

Variance reduction

$$\arg\min_{\rho} \left(C_\rho(x) - \log p(x|z)\right)^2$$

Generative gradient

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

Inference gradient

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

# 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`.

Evan Greensmith, Peter L Bartlett, and Jonathan Baxter. Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research*, 5(Nov):1471–1530, 2004.

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`.

Shixiang Gu, Sergey Levine, Ilya Sutskever, and Andriy Mnih. Muprop: Unbiased backpropagation for stochastic neural networks. *arXiv preprint arXiv:1511.05176*, 2015.

# Literature II

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`.

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`.