Stochastic Gradients of ELBO

Evgeny Burnaev

Skoltech, Moscow, Russia

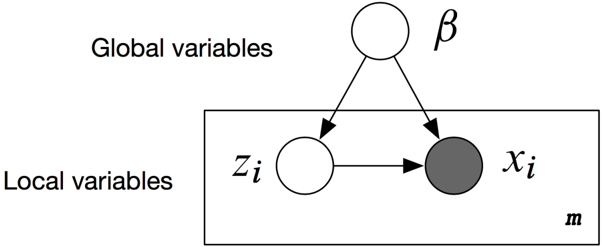
Skolkovo Institute of Science and Technology

Outline

1. Models with Latent Variables
2. Score Function Gradients of the ELBO
3. Pathwise Gradients of the ELBO
4. Amortized Inference

1. Models with Latent Variables
2. Score Function Gradients of the ELBO
3. Pathwise Gradients of the ELBO
4. Amortized Inference

A Generic Class of Models



$$p(\beta, \mathbf{Z}, \mathbf{X}) = p(\beta) \prod_{i=1}^{m} p(\mathbf{z}_i, \mathbf{x}_i | \beta)$$

- The observations are $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$
- The local variables are $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_m\}$
- The global variables are β
- The i-th data point $\mathbf{x}_i$ only depends on $\mathbf{z}_i$ and β
- Our aim:

$$\text{Compute } \; p(\beta, \mathbf{Z} | \mathbf{X})$$

- The observations are $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$
- The local variables are $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_m\}$
- Example: GMM with
  - $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$ — observations from

$$p(\mathbf{x}) = \sum_{k=1}^{K} p(\mathbf{z} = k|\beta) \cdot p(\mathbf{x}|\mathbf{z} = k, \beta)$$

    with $p(\mathbf{x}|\mathbf{z} = k, \beta) = \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$
  - Unknown latent variables $\mathbf{Z} = \{\mathbf{z}_1, \ldots, \mathbf{z}_m\}$ with a distribution $p(\mathbf{z} = k|\beta) = \pi_k,\ k = 1, \ldots, K$
  - Unknown parameters $\beta = \{\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{K}$

$$p(\beta, \mathbf{Z}, \mathbf{X}) = p(\beta) \prod_{i=1}^{m} p(\mathbf{z}_i, \mathbf{x}_i|\beta)$$

- Not to overburden slides with notations we consider just

$$p(\mathbf{x}, \mathbf{z})$$

- Our model — joint distribution of
  - observations $\mathbf{x}$
  - and latent variables $\mathbf{z}$
- Out aim is to estimate $p(\mathbf{z}|\mathbf{x})$
- Variational Bayes

$$q^* = \arg\min_{q \in Q} KL(q(\cdot)||p(\cdot|\mathbf{x}))$$

- Variational Evidence Lower Bound (ELBO)

$$KL(q(\cdot)||p(\cdot|\mathbf{x})) =$$
$$= \log p(\mathbf{x}) - \underbrace{\int q(\mathbf{z}) \log \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z})} d\mathbf{z}}_{\text{ELBO } \mathcal{L}(q)} \geq 0$$

- Thus $\log p(\mathbf{x}) \geq \mathcal{L}(q)$, and so we define

$$q^* = \arg\max_{q \in Q} \mathcal{L}(q)$$

- We start with a model $p(\mathbf{z}, \mathbf{x})$
- We choose a variational approximation $q(\mathbf{z}|\boldsymbol{\theta})$
- We write down the ELBO

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &= \int q(\mathbf{z}|\boldsymbol{\theta}) \log \frac{p(\mathbf{x}, \mathbf{z})}{q(\mathbf{z}|\boldsymbol{\theta})} d\mathbf{z} \\ &= \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\theta})}[\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta})] \to \max_{\boldsymbol{\theta}}\end{aligned}$$

- Data pairs $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$
- Inputs $\mathbf{x}_i$
- Output labels y_i
- $\mathbf{z}$ is a regression coefficient
- Generative process

Step 1: $p(\mathbf{z}) \sim \mathcal{N}(0, 1)$

Step 2: $p(y_i|\mathbf{x}_i, \mathbf{z}) \sim \mathrm{Bernoulli}(\sigma(\mathbf{z}\mathbf{x}_i)),\ i = 1, \ldots, m$

- Assume:
  - We have one data point $(y, \mathbf{x})$ $(m = 1)$
  - $\mathbf{x}$ is a scalar
  - The approximating family q is the normal, i.e.

$$q(\mathbf{z}|\boldsymbol{\theta}) = \mathcal{N}(\mathbf{z}|\mu, \sigma^2),\ \boldsymbol{\theta} = (\mu, \sigma)$$

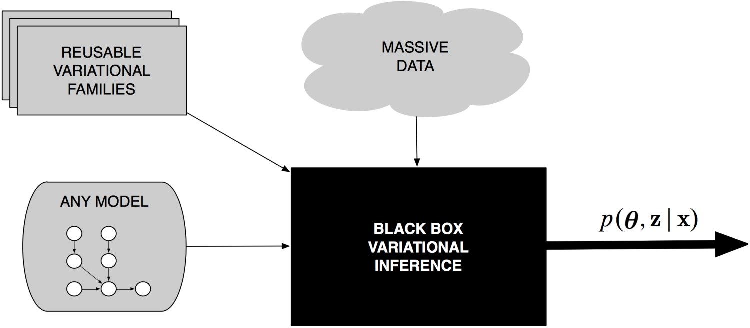
- The ELBO is

$$\begin{aligned}\mathcal{L}(\mu, \sigma^2) &= \mathbb{E}_{\mathbf{z}\sim q}\left[\log p(y, \mathbf{z}|\mathbf{x}) - \log q(\mathbf{z})\right] \\ &= \mathbb{E}_q\left[\log p(\mathbf{z}) + \log p(y|\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z})\right]\end{aligned}$$

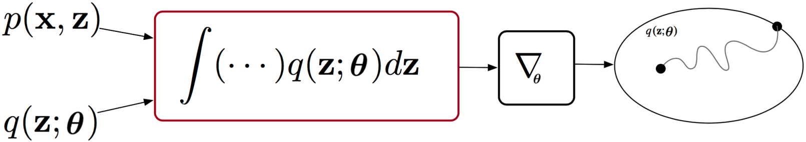
$$
\begin{aligned}
&\mathcal{L}(\mu, \sigma^2) = \\
&= \mathbb{E}_{\mathbf{z}\sim q}[\log p(\mathbf{z}) - \log q(\mathbf{z}) + \log p(y|\mathbf{x}, \mathbf{z})] \\
&= -\frac{1}{2}(\mu^2 + \sigma^2) + \frac{1}{2}\log \sigma^2 + \mathbb{E}_{\mathbf{z}\sim q}[\log p(y|\mathbf{x}, \mathbf{z})] + \mathrm{const} \\
&= -\frac{1}{2}(\mu^2 + \sigma^2) + \frac{1}{2}\log \sigma^2 + \mathbb{E}_{\mathbf{z}\sim q}[y\mathbf{x}\mathbf{z} - \log(1 + \exp(\mathbf{x}\mathbf{z}))] + \mathrm{const} \\
&= -\frac{1}{2}(\mu^2 + \sigma^2) + \frac{1}{2}\log \sigma^2 + y\mathbf{x}\mu - \mathbb{E}_{\mathbf{z}\sim q(\cdot|\boldsymbol{\theta})}[\log(1 + \exp(\mathbf{x}\mathbf{z}))] + \mathrm{const}
\end{aligned}
$$

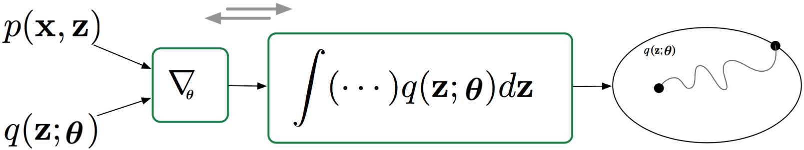
- We cannot analytically take that expectation
- The expectation hides the objectives dependence on the variational parameters $\boldsymbol{\theta} = (\mu, \sigma)$. This makes it hard to directly optimize

Black Box Variational Inference (BBVI)



The Problem in the Classical VI Recipe





Use stochastic optimization!

- Define

$$g(\mathbf{z}, \boldsymbol{\theta}) = \log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta})$$

- Gradient?

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}} \mathcal{L} &= \nabla_{\boldsymbol{\theta}} \int q(\mathbf{z}|\boldsymbol{\theta}) g(\mathbf{z}, \boldsymbol{\theta}) d\mathbf{z} \\
&= \int \left[\nabla_{\boldsymbol{\theta}} q(\mathbf{z}|\boldsymbol{\theta}) \cdot g(\mathbf{z}, \boldsymbol{\theta}) + q(\mathbf{z}|\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} g(\mathbf{z}, \boldsymbol{\theta})\right] d\mathbf{z} \\
&= \int \left[q(\mathbf{z}|\boldsymbol{\theta}) \frac{\nabla_{\boldsymbol{\theta}} q(\mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z}|\boldsymbol{\theta})} \cdot g(\mathbf{z}|\boldsymbol{\theta}) + q(\mathbf{z}|\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} g(\mathbf{z}, \boldsymbol{\theta})\right] d\mathbf{z} \\
&= \int \left[q(\mathbf{z}|\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta}) \cdot g(\mathbf{z}|\boldsymbol{\theta}) + q(\mathbf{z}|\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} g(\mathbf{z}, \boldsymbol{\theta})\right] d\mathbf{z} \\
&= \int q(\mathbf{z}|\boldsymbol{\theta}) \left[\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta}) \cdot g(\mathbf{z}|\boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}} g(\mathbf{z}, \boldsymbol{\theta})\right] d\mathbf{z} \\
&= \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\theta})} \left[\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta}) \cdot g(\mathbf{z}|\boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}} g(\mathbf{z}, \boldsymbol{\theta})\right]
\end{aligned}$$

Approaches

- Score Function Gradients
- Pathwise Gradients
- Amortized Inference

1 Models with Latent Variables

2 Score Function Gradients of the ELBO

3 Pathwise Gradients of the ELBO

4 Amortized Inference

- Recall

$$\nabla_{\boldsymbol{\theta}} \mathcal{L} = \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\theta})}[\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta}) g(\mathbf{z}, \boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}} g(\mathbf{z}, \boldsymbol{\theta})]$$

- We get that

$$\int q(\mathbf{z}|\boldsymbol{\theta}) d\mathbf{z} = 1 \Rightarrow \nabla_{\boldsymbol{\theta}} \int q(\mathbf{z}|\boldsymbol{\theta}) d\mathbf{z} = 0 \Rightarrow \int \nabla_{\boldsymbol{\theta}} q(\mathbf{z}|\boldsymbol{\theta}) d\mathbf{z} = 0$$

$$\int \frac{\nabla_{\boldsymbol{\theta}} q(\mathbf{z}|\boldsymbol{\theta})}{q(\mathbf{z}|\boldsymbol{\theta})} \cdot q(\mathbf{z}|\boldsymbol{\theta}) d\mathbf{z} = 0 \Rightarrow \int [\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})] \cdot q(\mathbf{z}|\boldsymbol{\theta}) d\mathbf{z} = 0$$

$$\mathbb{E}_q[\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})] = 0$$

- Since $g(\mathbf{z}, \boldsymbol{\theta}) = \log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta})$, then

$$\nabla_{\boldsymbol{\theta}} g(\mathbf{z}, \boldsymbol{\theta}) = -\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})$$

$$\mathbb{E}_q[\nabla_{\boldsymbol{\theta}} g(\mathbf{z}, \boldsymbol{\theta})] = -\mathbb{E}_q[\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})] = 0$$

- We get the gradient

$$\nabla_{\boldsymbol{\theta}} \mathcal{L} = \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\theta})}[\{\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})\} \cdot g(\mathbf{z}, \boldsymbol{\theta})]$$

$$\nabla_{\boldsymbol{\theta}} \mathcal{L} = \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\theta})}[\nabla_{\boldsymbol{\theta}} \{\log q(\mathbf{z}|\boldsymbol{\theta})\} \cdot (\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta}))]$$

Sometimes called likelihood ratio or REINFORCE gradients

- Gradient

$$\mathbb{E}_{q(\mathbf{z}|\boldsymbol{\theta})}[\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta}) \cdot (\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta}))]$$

- Noisy unbiased gradients with Monte Carlo!

$$\frac{1}{S} \sum_{s=1}^{S} \nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}_s|\boldsymbol{\theta}) \cdot (\log p(\mathbf{x}, \mathbf{z}_s) - \log q(\mathbf{z}_s|\boldsymbol{\theta})),$$

where $\mathbf{z}_s \sim q(\mathbf{z}|\boldsymbol{\theta})$

Basic Black Box Variational Inference

- **Input**: Model $\log p(\mathbf{x}, \mathbf{z})$, variational approximation $q(\mathbf{z}|\boldsymbol{\theta})$
- **Output**: Variational Parameters $\boldsymbol{\theta}$
- **while** not converged **do**
- $\mathbf{z}_s \sim q(\cdot|\boldsymbol{\theta})$ — Draw S samples from q
- $\rho = t$-th value of a Robbins Monro sequence
- We update

$$\boldsymbol{\theta} \Leftarrow \boldsymbol{\theta} + \rho \frac{1}{S} \sum_{s=1}^{S} \nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}_s|\boldsymbol{\theta}) \cdot (\log p(\mathbf{x}, \mathbf{z}_s) - \log q(\mathbf{z}_s|\boldsymbol{\theta}))$$

- **end**

- The noisy gradient:

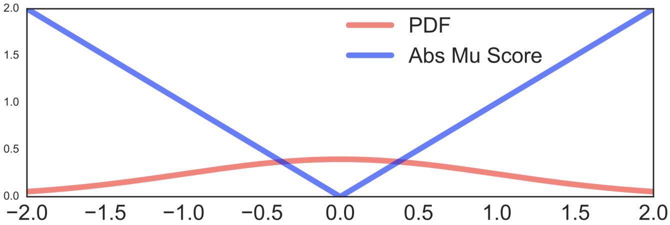
$$\frac{1}{S}\sum_{s=1}^{S} \nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}_s|\boldsymbol{\theta})(\log p(\mathbf{x}, \mathbf{z}_s) - \log q(\mathbf{z}_s|\boldsymbol{\theta})),$$

where $\mathbf{z}_s \sim q(\mathbf{z}|\boldsymbol{\theta})$

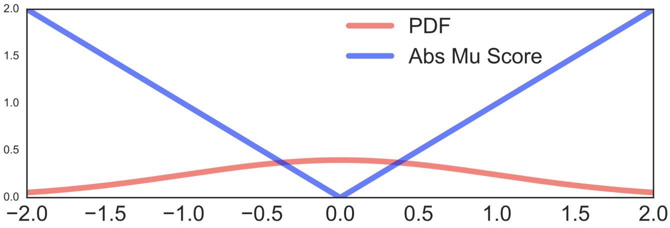
- To compute the noisy gradient of the ELBO we need
  - Sampling from $q(\mathbf{z}|\boldsymbol{\theta})$
  - Evaluating $\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})$
  - Evaluating $\log p(\mathbf{x}, \mathbf{z})$ and $\log q(\mathbf{z}|\boldsymbol{\theta})$
- There is no model specific work: black box criteria are satisfied!

Variance of the gradient can be a problem

$$\mathrm{Var}_{q(\mathbf{x}|\boldsymbol{\theta})} = \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\theta})}[(\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})(\log p(\mathbf{x},\mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta})) - \nabla_{\boldsymbol{\theta}}\mathcal{L})^2]$$



Intuition: Sampling rare values can lead to large scores and thus high variance



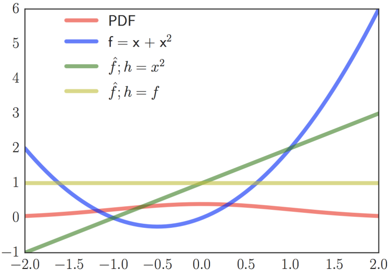
Replace f with $\widehat{f}$ where $\mathbb{E}[\widehat{f}(\mathbf{z})] = \mathbb{E}[f(\mathbf{z})]$. General class

$$\widehat{f}(\mathbf{z}) = f(\mathbf{z}) - a(h(\mathbf{z}) - \mathbb{E}[h(\mathbf{z})])$$

- For variational inference we need functions with known q expectation
- Set h as $\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})$
- Simple as $\mathbb{E}_q[\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})] = 0$ for any q

Replace f with $\widehat{f}$, where $\mathbb{E}[\widehat{f}(\mathbf{z})] = \mathbb{E}[f(\mathbf{z})]$.General class

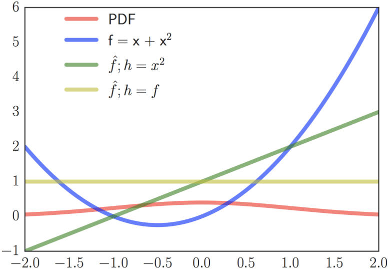
$$\widehat{f}(\mathbf{z}) = f(\mathbf{z}) - a(h(\mathbf{z}) - \mathbb{E}[h(\mathbf{z})])$$



- h is a function of our choice
- a is chosen to minimize the variance
- Good h have high correlation with the original function f

Replace f with $\widehat{f}$ where $EE[\widehat{f}(\mathbf{z})] = \mathbb{E}[f(\mathbf{z})]$. General class

$$\widehat{f}(\mathbf{z}) = f(\mathbf{z}) - a(h(\mathbf{z}) - \mathbb{E}[h(\mathbf{z})])$$

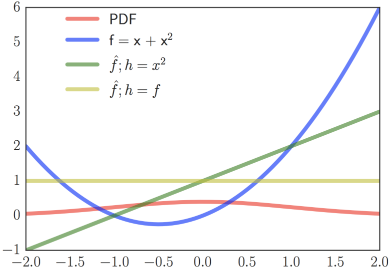


Many of the other techniques from Monte Carlo can help:

- For variational inference we need functions with known q expectation
- Set h as $\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})$
- Simple as $\mathbb{E}_q[\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})] = 0$ for any q

Replace f with $\widehat{f}$ where $\mathbb{E}[\widehat{f}(\mathbf{z})] = \mathbb{E}[f(\mathbf{z})]$. General class

$$\widehat{f}(\mathbf{z}) = f(\mathbf{z}) - a(h(\mathbf{z}) - \mathbb{E}[h(\mathbf{z})])$$



Many of the other techniques from Monte Carlo can help:

- Importance Sampling, Quasi Monte Carlo, Rao-Blackwellization

- The current black box criteria
  - Sampling from $q(\mathbf{z}|\boldsymbol{\theta})$
  - Evaluating $\nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta})$
  - Evaluating $\log p(\mathbf{x}, \mathbf{z})$ and $\log q(\mathbf{z}|\boldsymbol{\theta})$
- Can we make additional assumptions that are not too restrictive?

1 Models with Latent Variables

2 Score Function Gradients of the ELBO

3 Pathwise Gradients of the ELBO

4 Amortized Inference

Assume

1. Let $\mathbf{z} \sim q(\mathbf{z}|\boldsymbol{\theta})$ can be realized as $\mathbf{z} = t(\boldsymbol{\epsilon}, \boldsymbol{\theta})$ for some r.v. $\boldsymbol{\epsilon} \sim s(\boldsymbol{\epsilon})$. Example:

$$
\begin{aligned}
&\epsilon \sim \mathcal{N}(0, 1) \\
&z = \epsilon\sigma + \mu \quad \Rightarrow z \sim \mathcal{N}(z|\mu, \sigma^2)
\end{aligned}
$$

2. $\log p(\mathbf{x}, \mathbf{z})$ and $\log q(\mathbf{z}|\boldsymbol{\theta})$ are differentiable with respect to $\mathbf{z}$

- Let us for $\boldsymbol{\mu} \in \mathbb{R}^p$, $\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1^2, \dots, \sigma_p^2)$ set

$$q(\mathbf{z}|\boldsymbol{\theta}) = \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}, \boldsymbol{\Sigma}),\ \boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma})$$

- Thus

$$\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\Sigma}^{1/2}\boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}|\mathbf{0}, \mathrm{I})$$

- Since

$$\begin{aligned}
\log q(\mathbf{z}|\boldsymbol{\theta}) &= -\frac{1}{2}\log\det\boldsymbol{\Sigma} - \frac{1}{2}(\mathbf{z}-\boldsymbol{\mu})\boldsymbol{\Sigma}^{-1}(\mathbf{z}-\boldsymbol{\mu})^\top + \mathrm{const} \\
&= -\frac{1}{2}\log\prod_{i=1}^{p}\sigma_i^2 - \frac{1}{2}\sum_{i=1}^{p}\frac{(z_i-\mu_i)^2}{\sigma_i^2} + \mathrm{const} \\
&= -\sum_{i=1}^{p}\log\sigma_i - \frac{1}{2}\sum_{i=1}^{p}\frac{(z_i-\mu_i)^2}{\sigma_i^2} + \mathrm{const} \\
&= -\sum_{i=1}^{p}\log\sigma_i - \frac{1}{2}\sum_{i=1}^{p}\epsilon_i^2 + \mathrm{const}
\end{aligned}$$

- We would like to calculate $\nabla_{\boldsymbol{\theta}} \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\theta})}[\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta})]$
- We set $\mathbf{z} = \boldsymbol{\mu} + \boldsymbol{\Sigma}^{1/2}\boldsymbol{\epsilon}$ and

$$\log q(\mathbf{z}|\boldsymbol{\theta}) = -\sum_{i=1}^{p} \log \sigma_i - \frac{1}{2}\sum_{i=1}^{p} \epsilon_i^2 + \mathrm{const}$$

- E.g. for $\nabla_{\mu_j}\mathcal{L}(\boldsymbol{\theta})$ we get that

$$\begin{aligned}
\nabla_{\mu_j}\mathcal{L}(\boldsymbol{\theta}) &= \nabla_{\mu_j}\mathbb{E}_{\boldsymbol{\epsilon}}\left[\log p(\mathbf{x}, \boldsymbol{\mu} + \boldsymbol{\Sigma}^{1/2}\boldsymbol{\epsilon}) + \sum_{i=1}^{p} \log \sigma_i + \frac{1}{2}\sum_{i=1}^{p} \epsilon_i^2\right] \\
&= \mathbb{E}_{\boldsymbol{\epsilon}}\left[\nabla_{z_j} \log p(\mathbf{x}, \mathbf{z})\Big|_{\mathbf{z}=\boldsymbol{\mu}+\boldsymbol{\Sigma}^{1/2}\boldsymbol{\epsilon}} \cdot \nabla_{\mu_j}\left(\boldsymbol{\mu} + \boldsymbol{\Sigma}^{1/2}\boldsymbol{\epsilon}\right)\right] \\
&= \mathbb{E}_{\boldsymbol{\epsilon}}\left[\nabla_{z_j} \log p(\mathbf{x}, \mathbf{z})\Big|_{\mathbf{z}=\boldsymbol{\mu}+\boldsymbol{\Sigma}^{1/2}\boldsymbol{\epsilon}}\right] \\
&\approx \frac{1}{S}\sum_{s=1}^{S}\left[\nabla_{z_j} \log p(\mathbf{x}, \mathbf{z})\Big|_{\mathbf{z}_s=\boldsymbol{\mu}+\boldsymbol{\Sigma}^{1/2}\boldsymbol{\epsilon}_s}\right],
\end{aligned}$$

where $\boldsymbol{\epsilon}_s \sim \mathcal{N}(\boldsymbol{\epsilon}|\mathbf{0}, \mathrm{I})$

- Recall that for $g(\mathbf{z}, \boldsymbol{\theta}) = \log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta})$ we have

$$\nabla_{\boldsymbol{\theta}} \mathcal{L} = \nabla_{\boldsymbol{\theta}} \mathbb{E}_{q(\mathbf{z}|\boldsymbol{\theta})}[g(\mathbf{z}, \boldsymbol{\theta})]$$

- Rewrite using $\mathbf{z} = t(\boldsymbol{\epsilon}, \boldsymbol{\theta})$

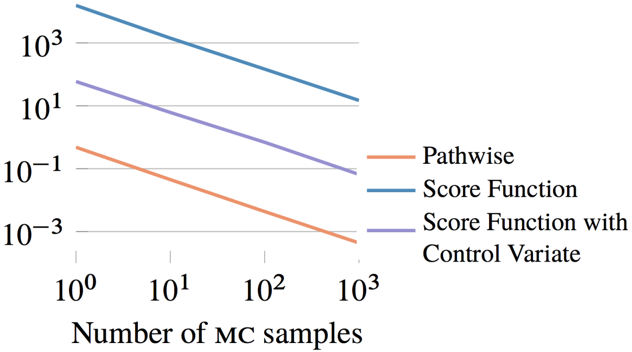
$$\nabla_{\boldsymbol{\theta}} \mathcal{L} = \nabla_{\boldsymbol{\theta}} \mathbb{E}_{\boldsymbol{\epsilon} \sim s(\boldsymbol{\epsilon})}[g(t(\boldsymbol{\epsilon}, \boldsymbol{\theta}), \boldsymbol{\theta})] = \mathbb{E}_{\boldsymbol{\epsilon} \sim s(\boldsymbol{\epsilon})}[\nabla_{\boldsymbol{\theta}} g(t(\boldsymbol{\epsilon}, \boldsymbol{\theta}), \boldsymbol{\theta})]$$

- We get that

$$\begin{aligned}
&\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_{s(\boldsymbol{\epsilon})}[\nabla_{\boldsymbol{\theta}} g(t(\boldsymbol{\epsilon}, \boldsymbol{\theta}), \boldsymbol{\theta})] \\
&= \mathbb{E}_{s(\boldsymbol{\epsilon})} \left[\nabla_{\mathbf{z}} g(\mathbf{z}, \boldsymbol{\theta}) \Big|_{\mathbf{z}=t(\boldsymbol{\epsilon}, \boldsymbol{\theta})} \cdot \nabla_{\boldsymbol{\theta}} t(\boldsymbol{\epsilon}, \boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}} g(\mathbf{z}, \boldsymbol{\theta}) \Big|_{\mathbf{z}=t(\boldsymbol{\epsilon}, \boldsymbol{\theta})} \right] \\
&= \mathbb{E}_{s(\boldsymbol{\epsilon})} \left[\nabla_{\mathbf{z}} [\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta})] \Big|_{\mathbf{z}=t(\boldsymbol{\epsilon}, \boldsymbol{\theta})} \cdot \nabla_{\boldsymbol{\theta}} t(\boldsymbol{\epsilon}, \boldsymbol{\theta}) - \nabla_{\boldsymbol{\theta}} \log q(\mathbf{z}|\boldsymbol{\theta}) \right] \\
&= \mathbb{E}_{s(\boldsymbol{\epsilon})} \left[\nabla_{\mathbf{z}} [\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta})] \Big|_{\mathbf{z}=t(\boldsymbol{\epsilon}, \boldsymbol{\theta})} \cdot \nabla_{\boldsymbol{\theta}} t(\boldsymbol{\epsilon}, \boldsymbol{\theta}) \right]
\end{aligned}$$

This is also known as the reparameterization gradient

Variance Comparison



[Kucukelbir+ 2016]

Score Function

- Differentiates the density $\nabla_{\boldsymbol{\theta}} q(\mathbf{z}|\boldsymbol{\theta})$
- Works for discrete and continuous models
- Works for large class of variational approximations
- Variance can be a big problem

Pathwise

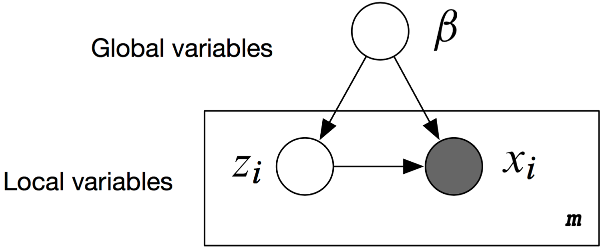
- Differentiates the function $\nabla_{\mathbf{z}}[\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}|\boldsymbol{\theta})]$
- Requires differentiable models
- Requires variational approximation to have the form $\mathbf{z} = t(\boldsymbol{\epsilon}, \boldsymbol{\theta})$
- Generally better behaved variance

1 Models with Latent Variables

2 Score Function Gradients of the ELBO

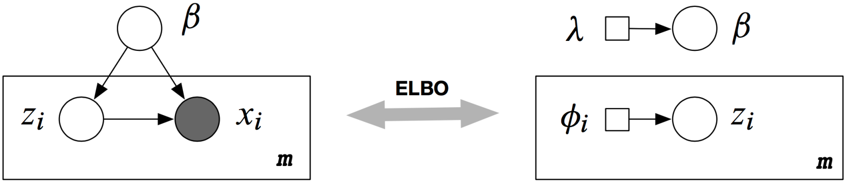
3 Pathwise Gradients of the ELBO

4 Amortized Inference



$$p(\beta, \mathbf{Z}, \mathbf{X}) = p(\beta) \prod_{i=1}^{m} p(\mathbf{z}_i, \mathbf{x}_i | \beta)$$

- The observations are $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_m\}$
- The local variables are $\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_m\}$
- The global variables are β


β
z_i
x_i
m
ELBO
λ
β
ϕ_i
z_i
m

- **Input**: data $\mathbf{X}$, model $p(\beta, \mathbf{Z}, \mathbf{X})$
- **Aim**: approximate the posterior $p(\beta, \mathbf{Z}|\mathbf{X})$
- The mean-field family for $\boldsymbol{\theta} = (\lambda, \phi_{1\ldots m})$

$$q(\beta, \mathbf{Z}|\boldsymbol{\theta}) = q(\beta|\lambda)\prod_{i=1}^{m} q(\mathbf{z}_i|\phi_i)$$

- The ELBO has the form

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &= \mathbb{E}_{q(\beta,\mathbf{Z}|\boldsymbol{\theta})}[\log p(\beta, \mathbf{Z}, \mathbf{X}) - \log q(\beta, \mathbf{Z}|\boldsymbol{\theta})] \\ &= \mathbb{E}_q[\log p(\beta, \mathbf{Z}, \mathbf{X})] - \mathbb{E}_q\left[\log q(\beta|\lambda) + \sum_{i=1}^{m}\log q(\mathbf{z}_i|\phi_i)\right]\end{aligned}$$

- These expectations are no longer tractable
- Inner stochastic optimization needed for each data point
- **Idea**: Learn a mapping f from $\mathbf{x}_i$ to ϕ_i!!!

- ELBO

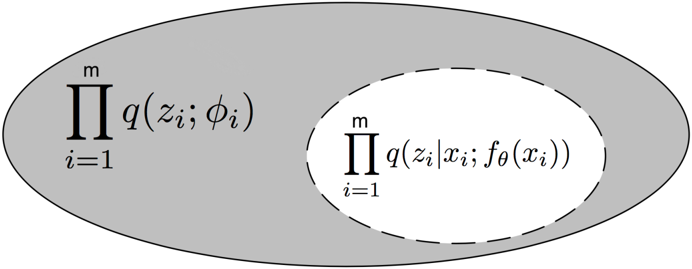
$$\mathcal{L}(\boldsymbol{\theta}) = \mathbb{E}_q[\log p(\beta, \mathbf{Z}, \mathbf{X})] - \mathbb{E}_q\left[\log q(\beta|\lambda) + \sum_{i=1}^{m} q(\mathbf{z}_i|\phi_i)\right]$$

- Amortizing the ELBO with inference network f:

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) = {} & \mathbb{E}_q[\log p(\beta, \mathbf{Z}, \mathbf{X})] - \\ & - \mathbb{E}_q\left[\log q(\beta|\lambda) + \sum_{i=1}^{m} \log q(\mathbf{z}_i|\mathbf{x}_i; \phi_i = f_\theta(\mathbf{x}_i))\right],\end{aligned}$$

here $\boldsymbol{\theta} = (\lambda, \theta)$

- Amortized inference is faster, but admits a smaller class of approximations
- The size of the smaller class depends on the flexibility of f



- If $\log p(\mathbf{x}, \mathbf{z})$ is differentiable w.r.t. $\mathbf{z}$
  - Try out an approximation q that is reparameterizable
- If $\log p(\mathbf{x}, \mathbf{z})$ is not differentiable w.r.t. $\mathbf{z}$
  - Use score function estimator with control variates
  - Add further variance reductions based on experimental evidence
- General Advice:
  - Use coordinate specific learning rates (e.g. RMSProp, AdaGrad)
  - Annealing + Tempering
  - Consider parallelizing across samples from q

- Systems with Variational Inference:
  - Venture, WebPPL, Edward, Stan, PyMC3, Infer.net, Anglican
    Good for trying out lots of models
- Differentiation Tools:
  - Theano, Torch, Tensorflow, Stan Math, Caffe

  Can lead to more scalable implementations of individual models