Extending the Reparameterization Trick

Michael Figurnov

Research Scientist at  DeepMind

August 30th, 2018

Take-away message

What you will learn from this lecture

- Stochastic gradient estimation is a general and important problem
  - Important for Bayesian deep learning, but also in other areas
  - Reparameterization and REINFORCE are two methods for this problem
  - There are many transferable ideas, including control variates

- Reparameterization can be applied to almost any continuous distribution
  - Normal
  - But also Gamma, Beta, Dirichlet, von Mises...

Outline

- **Overview**
  - Stochastic Gradient Estimation
  - REINFORCE
  - Reparameterization gradients
  - Comparison
- Expanding the applicability of reparameterization
  - Generalized Reparameterization Gradients
  - Implicit Reparameterization Gradients
- Reducing the variance of the reparameterization gradient
  - Pathwise Derivatives for Multivariate Distributions

Stochastic gradient estimators

Some definitions

Parametric *continuous* "well-behaved" distribution $q_\phi(\boldsymbol{z}),\ \boldsymbol{z} \in \mathbb{R}^D,\ \phi \in \mathbb{R}$

differentiable density, no inner regions of zero density

Differentiable cost function $f(\boldsymbol{z}) \in \mathbb{R}$

Expected/mean cost $\mathcal{L}(\phi) = \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[f(\boldsymbol{z})\right]$

Goal: unbiased stochastic estimator for $\mathcal{L}'(\phi)$:

$$\mathcal{L}'(\phi) = \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[g(\boldsymbol{z}, \phi)\right] \approx \frac{1}{M}\sum_{i=1}^{M} g(\boldsymbol{z}_i, \phi),\ \boldsymbol{z}_i \sim q_\phi(\boldsymbol{z})$$

Bonus points: an estimator with low variance $\mathrm{Var}_{q_\phi(\boldsymbol{z})}\left[g(\boldsymbol{z}, \phi)\right]$

Applications of stochastic gradient estimators

- Machine Learning
  - Bayesian Deep Learning
    - Variational Autoencoders
    - Variational Dropout
    - Bayesian Neural Networks
  - Reinforcement Learning
    - Policy gradient methods
    - Natural Evolution Strategies
- Non-Machine Learning
  - Computational Finance
    - Sensitivity analysis
  - Operation research
    - Queuing theory

ELBO $\mathcal{L}(\phi) = \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[\log \frac{p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})}{q_\phi(\boldsymbol{z})}\right]$

expected reward $\mathcal{L}(\phi) = \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[r(\boldsymbol{z})\right]$

Fu "Gradient estimation." Handbooks in operations research and management science 2006
Glasserman “Monte Carlo methods in financial engineering.” Springer 2013

Reminder: control variates

General way of reducing the variance of stochastic estimators

Consider $h(\boldsymbol{z}, \phi)$ that is correlated with $g(\boldsymbol{z}, \phi)$ and has tractable expectation

$\tilde{g}(\boldsymbol{z}, \phi) = g(\boldsymbol{z}, \phi) - \gamma \left(h(\boldsymbol{z}, \phi) - \mathbb{E}_{q_\phi(\boldsymbol{z})} \left[h(\boldsymbol{z}, \phi) \right] \right)$ is an unbiased estimator

$$\operatorname{Var}\left[\tilde{g}(\boldsymbol{z}, \phi)\right] = \operatorname{Var}\left[g(\boldsymbol{z}, \phi)\right] - \gamma \operatorname{Cov}\left[g(\boldsymbol{z}, \phi), h(\boldsymbol{z}, \phi)\right] + \gamma^2 \operatorname{Var}\left[h(\boldsymbol{z}, \phi)\right]$$

Minimal variance: $$\gamma^* = \frac{\operatorname{Cov}\left[g(\boldsymbol{z}, \phi), h(\boldsymbol{z}, \phi)\right]}{\operatorname{Var}\left[h(\boldsymbol{z}, \phi)\right]}$$

Glasserman “Monte Carlo methods in financial engineering.” Springer 2013, Chapter 4.1.1

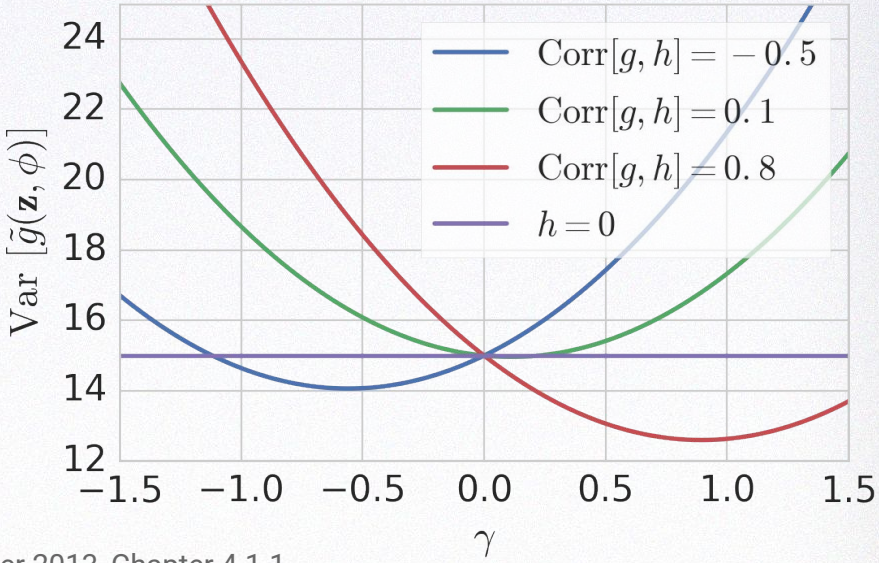
Reminder: control variates

General way of reducing the variance of stochastic estimators

$$\mathrm{Var}\left[\tilde{g}(\boldsymbol{z},\phi)\right] = \mathrm{Var}\left[g(\boldsymbol{z},\phi)\right] - \gamma\,\mathrm{Cov}\left[g(\boldsymbol{z},\phi), h(\boldsymbol{z},\phi)\right] + \gamma^2\,\mathrm{Var}\left[h(\boldsymbol{z},\phi)\right]$$

Min variance: $\gamma^* = \dfrac{\mathrm{Cov}\left[g(\boldsymbol{z},\phi), h(\boldsymbol{z},\phi)\right]}{\mathrm{Var}\left[h(\boldsymbol{z},\phi)\right]}$

If γ is not optimal, variance can *increase*!



Glasserman “Monte Carlo methods in financial engineering.” Springer 2013, Chapter 4.1.1

REINFORCE gradient estimator

aka Score Function estimator, Likelihood Ratio estimator

$$\mathcal{L}'(\phi) = \mathbb{E}_{q_\phi(z)}\underbrace{\left[f(z)\frac{\partial \log q_\phi(z)}{\partial \phi}\right]}_{g(z,\phi)}$$

- Does not require gradients of $f(z)$
- **Usually high variance**

```
z = tf.stop_gradient(q.sample())
surrogate_loss = f(z) * q.log_prob(z)
loss = tf.stop_gradient(f(z) - surrogate_loss) + surrogate_loss
```

value is $f(z)$, derivative is $g(z, \phi)$

Williams “Simple statistical gradient-following algorithms for connectionist reinforcement learning”. Machine Learning, 1992

Example: Normal distribution $z \sim \mathcal{N}(\mu, \sigma^2)$

Log-density: $\log q_{\mu,\sigma}(z) = -\frac{(z-\mu)^2}{2\sigma^2} - \log \sigma - \frac{1}{2}\log 2\pi$

REINFORCE gradient estimator:

$$g_\mu(z, \mu, \sigma) = f(z)\frac{\partial \log q_{\mu,\sigma}(z)}{\partial \mu} = f(z)\frac{z-\mu}{\sigma^2}$$

$$g_\sigma(z, \mu, \sigma) = f(z)\frac{\partial \log q_{\mu,\sigma}(z)}{\partial \sigma} = f(z)\frac{(z-\mu)^2 - \sigma^2}{\sigma^3}$$

Williams “Simple statistical gradient-following algorithms for connectionist reinforcement learning”. Machine Learning, 1992

Control variate for REINFORCE (baseline)

$$h(\boldsymbol{z}, \phi) = \frac{\partial \log q_\phi(\boldsymbol{z})}{\partial \phi}, \quad \mathbb{E}_{q_\phi(\boldsymbol{z})}\, h(\boldsymbol{z}, \phi) = \frac{\partial}{\partial \phi}\, \mathbb{E}_{q_\phi(\boldsymbol{z})}\, 1 = 0$$

$$\tilde{g}(\boldsymbol{z}, \phi) = (f(\boldsymbol{z}) - \gamma)\frac{\partial \log q_\phi(\boldsymbol{z})}{\partial \phi}$$

is an unbiased estimator of $\mathcal{L}'(\phi)$

Williams “Simple statistical gradient-following algorithms for connectionist reinforcement learning”. Machine Learning, 1992

Control variate for REINFORCE (baseline)

$$\tilde{g}(\boldsymbol{z}, \phi) = (f(\boldsymbol{z}) - \gamma) \frac{\partial \log q_\phi(\boldsymbol{z})}{\partial \phi}$$

How to choose γ ?

- Mean cost $\gamma \approx \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[f(\boldsymbol{z})\right]$, easy to estimate, but can *increase* variance
- Optimal $\gamma^* = \dfrac{\mathrm{Cov}\left[f(\boldsymbol{z}) \frac{\partial \log q_\phi(\boldsymbol{z})}{\partial \phi}, \frac{\partial \log q_\phi(\boldsymbol{z})}{\partial \phi}\right]}{\mathrm{Var}\left[\frac{\partial \log q_\phi(\boldsymbol{z})}{\partial \phi}\right]}$, harder to estimate

Williams “Simple statistical gradient-following algorithms for connectionist reinforcement learning”. Machine Learning, 1992
Glasserman “Monte Carlo methods in financial engineering.” Springer 2013, Chapter 4.1.1

Reparameterization gradient estimator

Example: Normal distribution $z \sim \mathcal{N}(\mu, \sigma^2)$

Standardization: $\mathcal{S}_{\mu,\sigma}(z) = (z - \mu)/\sigma = \varepsilon \sim \mathcal{N}(0, 1)$

Inverse: $$z = \mathcal{S}^{-1}_{\mu,\sigma}(\varepsilon) = \mu + \sigma\varepsilon, \quad \frac{\partial z}{\partial \mu} = 1, \quad \frac{\partial z}{\partial \sigma} = \varepsilon = \frac{z - \mu}{\sigma}$$

Reparameterization gradient estimator:

$$g_\mu(z, \mu, \sigma) = f'(z)$$

$$g_\sigma(z, \mu, \sigma) = f'(z)\frac{z - \mu}{\sigma}$$

Williams “Simple statistical gradient-following algorithms for connectionist reinforcement learning”. Machine Learning, 1992
Kingma, Welling “Auto-encoding variational bayes”, ICLR 2014
Rezende, Mohamed, Wierstra “Stochastic backpropagation and approximate inference in deep generative models”, ICML 2014

Reparameterization gradient estimator

aka pathwise gradients

Standardization function: $\mathcal{S}_\phi(\boldsymbol{z}) = \boldsymbol{\varepsilon} \sim q(\boldsymbol{\varepsilon})$

Invert the function: $\boldsymbol{z} = \mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon})$

Reparameterization: $\mathbb{E}_{q_\phi(\boldsymbol{z})}\left[f(\boldsymbol{z})\right] = \mathbb{E}_{q(\boldsymbol{\varepsilon})}\left[f(\mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon}))\right]$

Move the gradient inside:

$$\frac{\partial}{\partial\phi}\mathbb{E}_{q_\phi(\boldsymbol{z})}\left[f(\boldsymbol{z})\right] = \mathbb{E}_{q(\boldsymbol{\varepsilon})}\left[\frac{\partial f(\mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon}))}{\partial\phi}\right]$$

$$= \mathbb{E}_{q(\boldsymbol{\varepsilon})}\left[\frac{\partial f(\mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon}))}{\partial\mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon})}\frac{\partial\mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon})}{\partial\phi}\right]$$

Kingma, Welling “Auto-encoding variational bayes”, ICLR 2014
Rezende, Mohamed, Wierstra “Stochastic backpropagation and approximate inference in deep generative models”, ICML 2014

Reparameterization gradient estimator

One more step to make it a stochastic gradient estimator

Change of variable $\boldsymbol{z} = \mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon})$:

$$\frac{\partial}{\partial \phi}\mathbb{E}_{q_\phi(\boldsymbol{z})}\left[f(\boldsymbol{z})\right] = \mathbb{E}_{q(\boldsymbol{\varepsilon})}\left[\frac{\partial f(\mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon}))}{\partial \mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon})}\frac{\partial \mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon})}{\partial \phi}\right]$$

$$= \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[\frac{\partial f(\boldsymbol{z})}{\partial \boldsymbol{z}}\frac{\partial \boldsymbol{z}}{\partial \phi}\right], \quad \frac{\partial \boldsymbol{z}}{\partial \phi} = \left.\frac{\partial \mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon})}{\partial \phi}\right|_{\boldsymbol{\varepsilon}=\mathcal{S}_\phi(\boldsymbol{z})}$$

$$g(\boldsymbol{z}, \phi) = \underset{1\times D}{\frac{\partial f(\boldsymbol{z})}{\partial \boldsymbol{z}}}\underset{D\times 1}{\frac{\partial \boldsymbol{z}}{\partial \phi}}$$

$\boldsymbol{z}$ is a “differentiable stochastic layer”: perfect for deep learning frameworks!

Figurnov, Mohamed, Mnih “Implicit Reparameterization Gradients”, 2018

Comparison of the estimators

REINFORCE vs. Reparameterization

$$\mathcal{L}(\mu, \sigma) = \mathbb{E}_{\mathcal{N}(z|\mu,\sigma^2)}\left[z^2\right]$$

	$\frac{\partial \mathcal{L}(\mu,\sigma)}{\partial \mu}$	$\frac{\partial \mathcal{L}(\mu,\sigma)}{\partial \sigma}$
REINFORCE	$z^2 \frac{z-\mu}{\sigma^2} = O(z^3)$	$z^2 \frac{(z-\mu)^2 - \sigma^2}{\sigma^3} = O(z^4)$
Reparameterization	$2z = O(z)$	$2z\frac{z-\mu}{\sigma} = O(z^2)$

In this case, REINFORCE estimator is a higher order polynomial

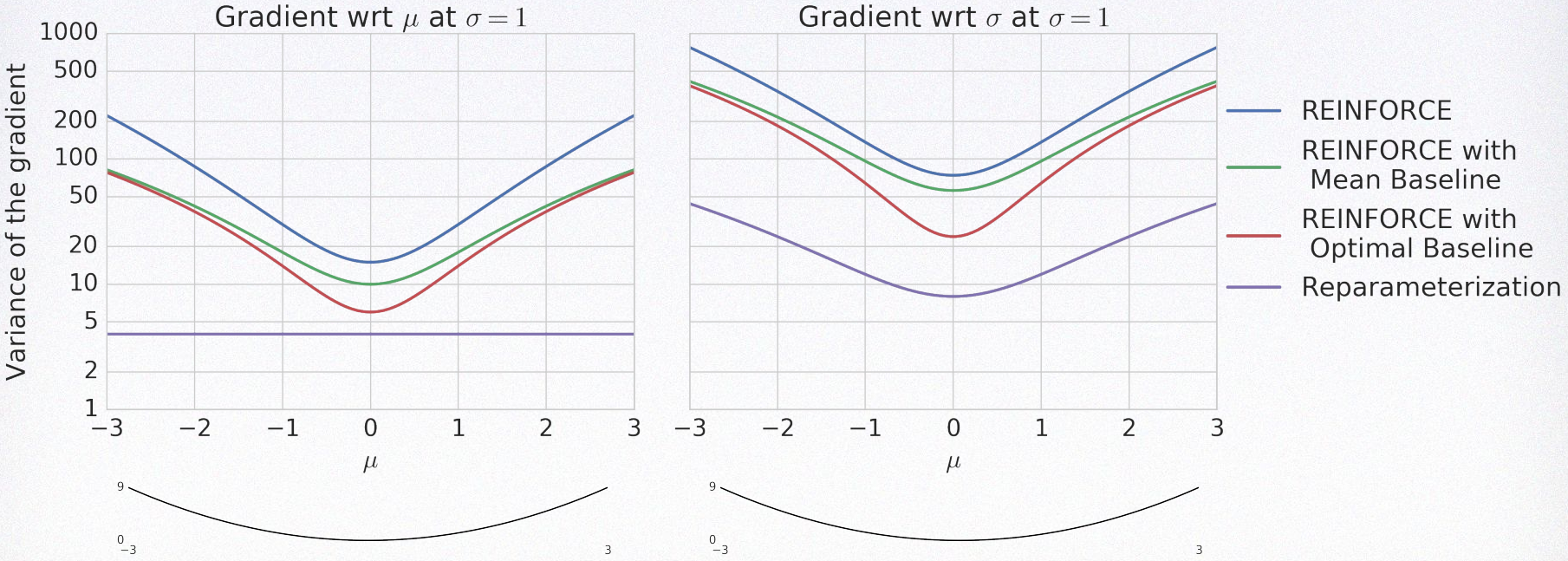
$\Rightarrow$ much larger values for $|z| > 1$

$\Rightarrow$ higher variance

Comparison of the estimators

REINFORCE vs. Reparameterization

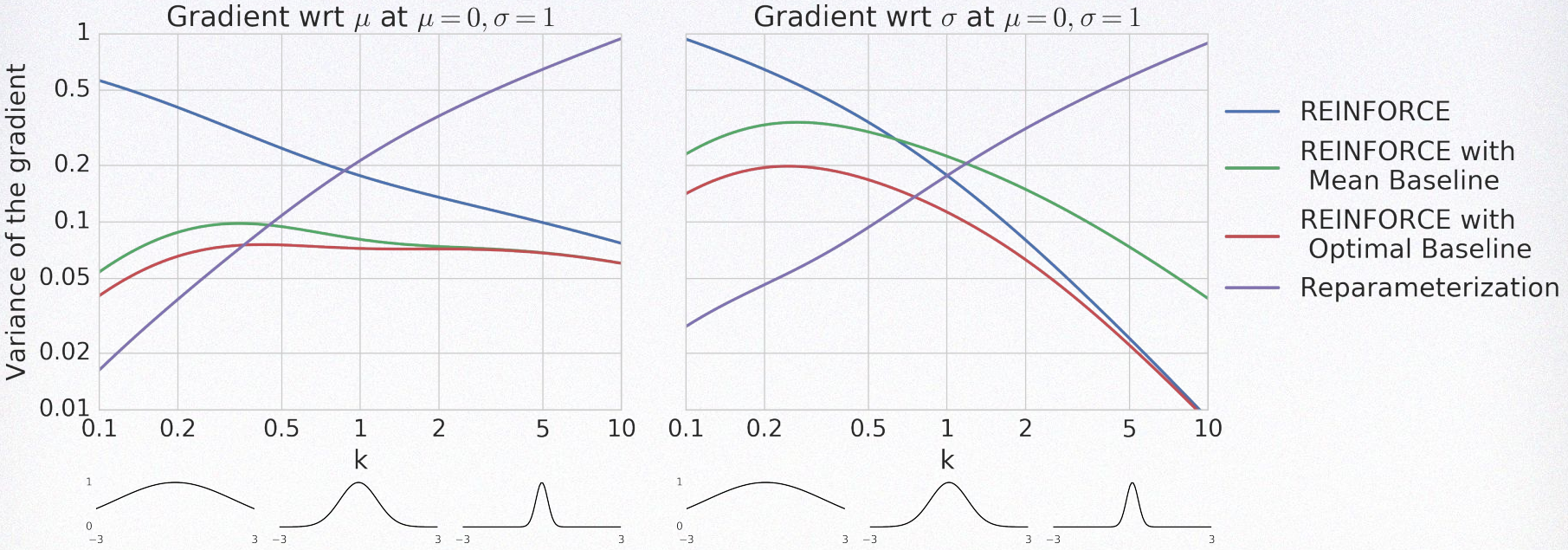
$$\mathcal{L}(\mu, \sigma) = \mathbb{E}_{\mathcal{N}(z|\mu,\sigma^2)}\left[z^2\right]$$



Comparison of the estimators

REINFORCE vs. Reparameterization

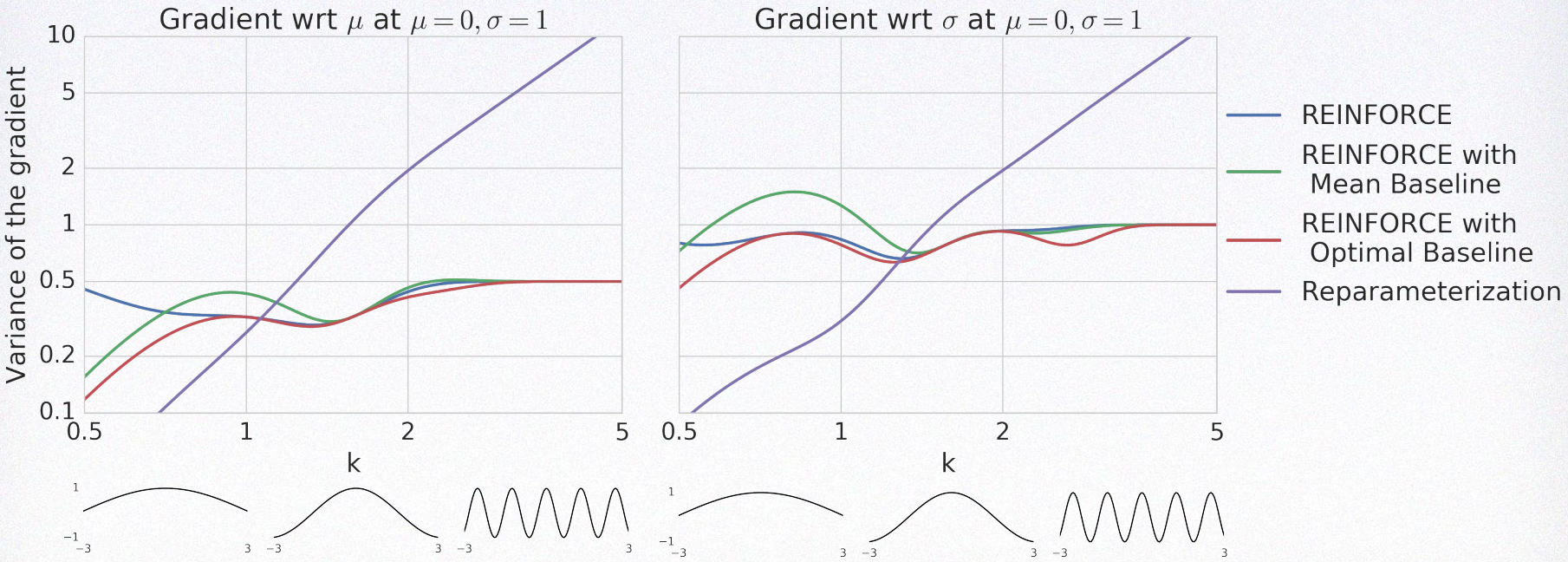
$$\mathcal{L}(\mu, \sigma) = \mathbb{E}_{\mathcal{N}(z|\mu,\sigma^2)}\left[\exp(-kz^2)\right]$$



Comparison of the estimators

REINFORCE vs. Reparameterization

$$\mathcal{L}(\mu, \sigma) = \mathbb{E}_{\mathcal{N}(z|\mu,\sigma^2)}\left[\cos kz\right]$$



Reparameterization gradients issues

- Not applicable to discrete distributions
  - Relaxation methods; REBAR
  - “Discrete Latent Variables” lecture on Tuesday

- **Limited scope of applicability**
  - Distributions with simple inverse of the standardization function

- **The variance of the estimator may be high**
  - Need a good control variate

Outline

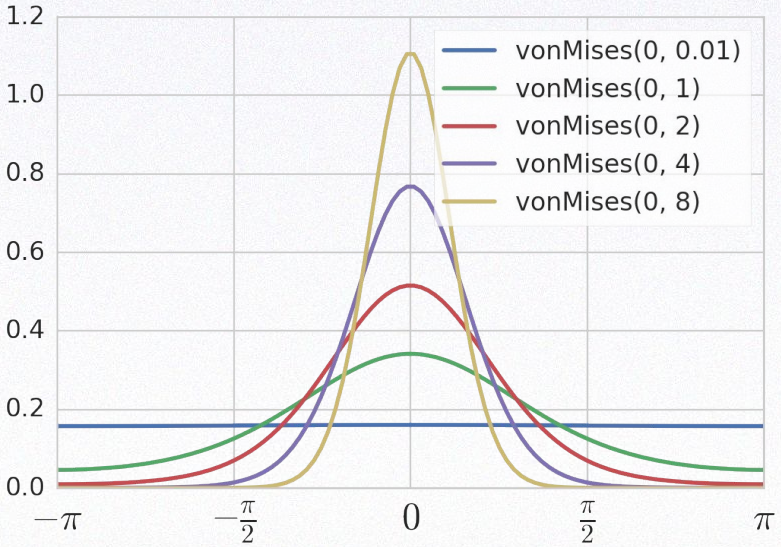
- Overview
  - Stochastic Gradient Estimation
  - REINFORCE
  - Reparameterization gradients
  - Comparison
- **Expanding the applicability of reparameterization**
  - Generalized Reparameterization Gradients
  - Implicit Reparameterization Gradients
- Reducing the variance of the reparameterization gradient
  - Pathwise Derivatives for Multivariate Distributions

Some hard to reparameterize distributions

Von Mises, distribution over angles (directions)

$$\text{vonMises}(z|\mu,\kappa) = \frac{\exp(\kappa\cos(z-\mu))}{2\pi I_0(\kappa)}$$

$$z \sim \text{vonMises}(0,\kappa) \Rightarrow z+\mu \sim \text{vonMises}(\mu,\kappa)$$

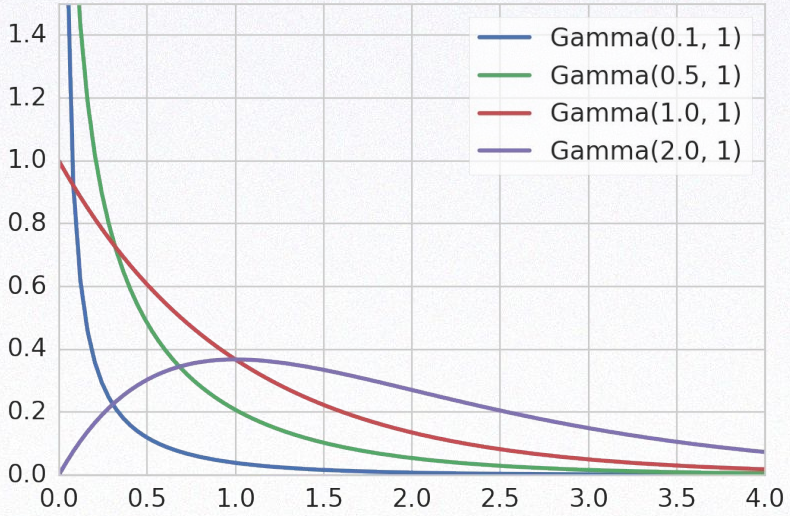


Some hard to reparameterize distributions

Gamma, conjugate prior for Normal

$$\text{Gamma}(z|\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} z^{\alpha-1}\exp(-\beta z)$$

$$z \sim \text{Gamma}(\alpha,1) \Rightarrow \frac{z}{\beta} \sim \text{Gamma}(\alpha,\beta)$$

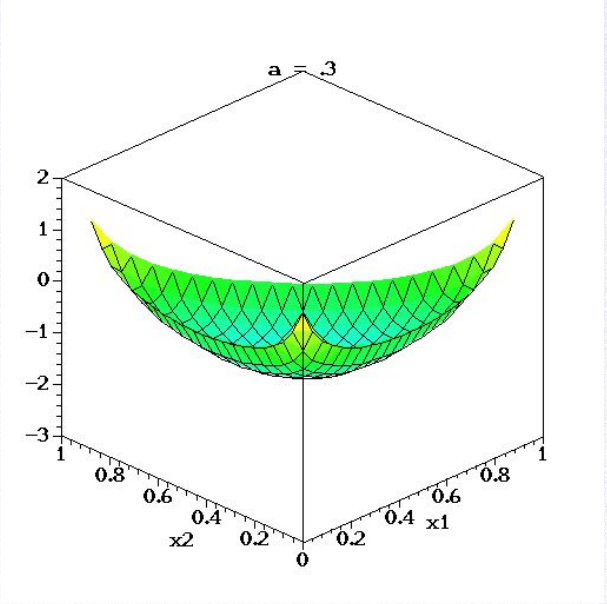


Some hard to reparameterize distributions

Dirichlet distribution, “sparse” distribution on simplex

$$\boldsymbol{z} \sim \text{Dirichlet}(\alpha_1, \ldots, \alpha_D), \;\; \alpha_i > 0$$

$$t_i \sim \text{Gamma}(\alpha_i, 1), \quad z_i = \frac{t_i}{\sum_{j=1}^{D} t_j}$$



Log-density of Dirichlet

Figure from Wikipedia, https://commons.wikimedia.org/wiki/File:LogDirichletDensity-alpha_0.3_to_alpha_2.0.gif

Outline

- Overview
  - Stochastic Gradient Estimation
  - REINFORCE
  - Reparameterization gradients
  - Comparison
- Expanding the applicability of reparameterization
  - **Generalized Reparameterization Gradients**
  - Implicit Reparameterization Gradients
- Reducing the variance of the reparameterization gradient
  - Pathwise Derivatives for Multivariate Distributions

Generalized Reparameterization Gradient

Motivation

Suppose that we cannot compute $u_q(\boldsymbol{z}, \phi) = \dfrac{\partial \boldsymbol{z}}{\partial \phi}$ for $q_\phi(\boldsymbol{z})$

...but we can compute $u_r(\boldsymbol{z}, \phi) = \dfrac{\partial \boldsymbol{z}}{\partial \phi}$ for a similar distribution $r_\phi(\boldsymbol{z})$

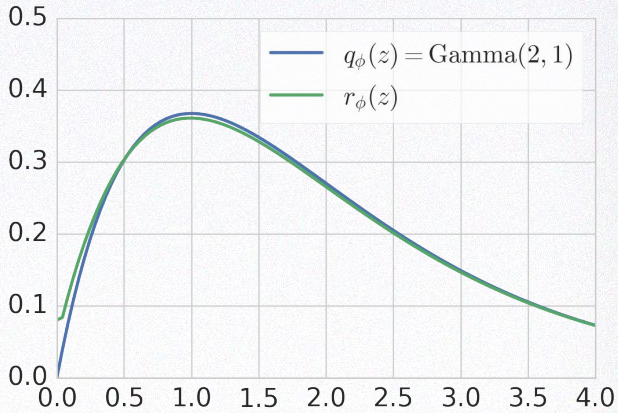
Idea: use reparameterization gradients for $r_\phi(\boldsymbol{z})$ and a correction term for bias

$$\mathcal{L}'(\phi) = \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[f(\boldsymbol{z}) \frac{\partial}{\partial \phi} \log \frac{q_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})} + \frac{\partial f(\boldsymbol{z})}{\partial \boldsymbol{z}} u_r(\boldsymbol{z}, \phi) \right]$$

REINFORCE-like correction

reparameterization gradient for $r_\phi(\boldsymbol{z})$

Alternative view: REINFORCE + baseline

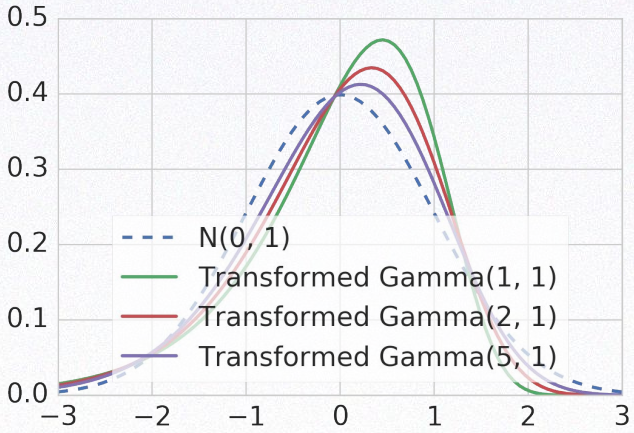


Ruiz, Titsias, Blei "The generalized reparameterization gradient." NIPS 2016

Naesseth, Ruiz, Linderman, Blei "Reparameterization gradients through acceptance-rejection sampling algorithms." AISTATS 2017

How to choose the approximating distribution?

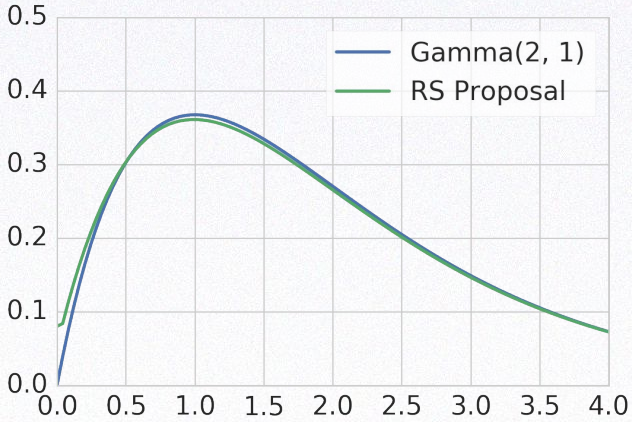
- *Generalized reparameterization gradients*
  - Apply a “partial standardization” to the samples (e.g., standardize the first and second moments)



Ruiz, Titsias, Blei "The generalized reparameterization gradient." NIPS 2016

How to choose the approximating distribution?

- *Reparameterization gradients through rejection sampling algorithms*
  - Use the proposal distribution of a RS method
  - Usually, a simple transformation of Uniform or Normal random variable

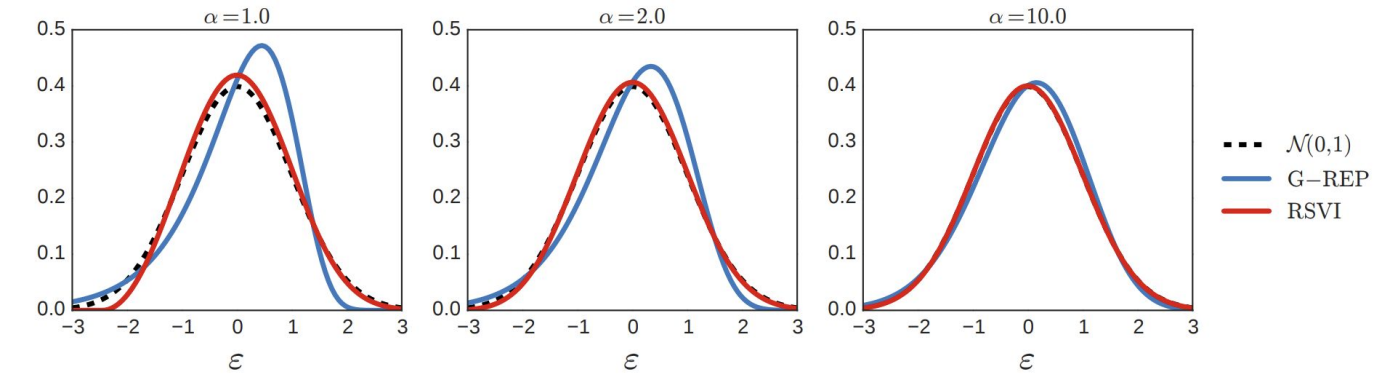


Naesseth, Ruiz, Linderman, Blei "Reparameterization gradients through acceptance-rejection sampling algorithms." AISTATS 2017

Shape augmentation trick for Gamma

Reducing the variance of the rejection sampling reparameterization gradients

- The rejection sampling proposal becomes exact for $\alpha \to \infty$



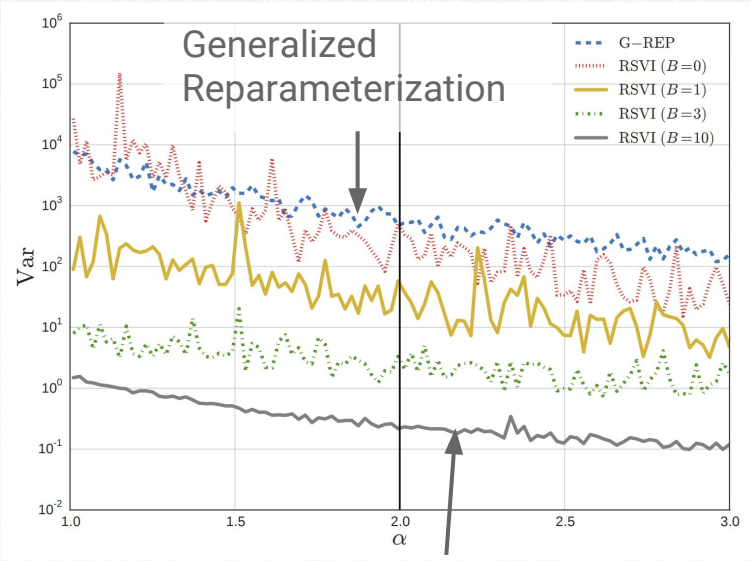
- Reparameterize $\mathrm{Gamma}(\alpha + B, 1)$, B - shape augmentation parameter
- $z \sim \mathrm{Gamma}(\alpha + 1, 1) \Rightarrow z u^{\frac{1}{\alpha}} \sim \mathrm{Gamma}(\alpha, 1),\ u \sim \mathrm{Uniform}(0, 1)$

Naesseth, Ruiz, Linderman, Blei "Reparameterization gradients through acceptance-rejection sampling algorithms." AISTATS 2017

Results

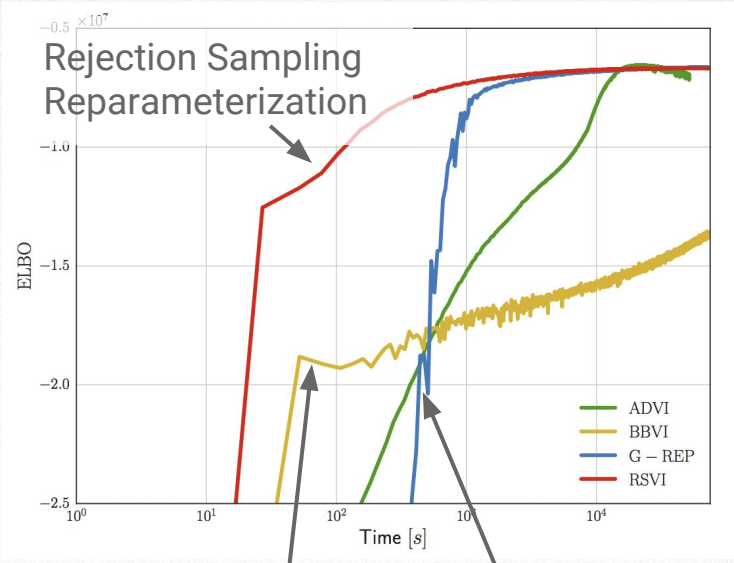
Rejection Sampling > Generalized Reparameterization > REINFORCE

Gradient of cross-entropy (synthetic problem)



Rejection Sampling Reparameterization
(B - shape augmentation)

Training a generative model of images



REINFORCE +
control variates

Generalized
Reparameterization

Naesseth, Ruiz, Linderman, Blei "Reparameterization gradients through acceptance-rejection sampling algorithms." AISTATS 2017

Questions & 5 minute break

Outline

- Overview
  - Stochastic Gradient Estimation
  - REINFORCE
  - Reparameterization gradients
  - Comparison
- Expanding the applicability of reparameterization
  - Generalized Reparameterization Gradients
  - **Implicit Reparameterization Gradients**
- Reducing the variance of the reparameterization gradient
  - Pathwise Derivatives for Multivariate Distributions

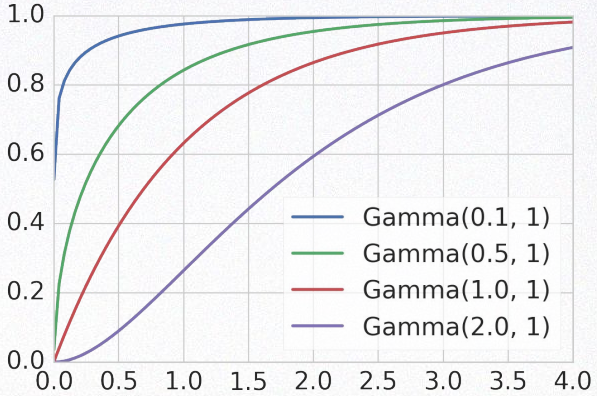
Implicit reparameterization gradients

Motivation

- A standardization function for univariate continuous distributions is CDF:

$$\mathcal{S}_\phi(z) = F(z|\phi) = \int_{-\infty}^{z} q_\phi(t)dt \sim \text{Uniform}(0,1)$$

- Computing the inverse and its derivative is hard
- Example: Gamma distribution $F(z|\alpha) = \frac{1}{\Gamma(\alpha)} \int_0^z t^{\alpha-1} \exp(-t)dt$
  - Intractable integral
  - Inversion via root-finding iterative methods (slow)
  - Derivative of the inverse via finite difference of iterative methods (poor precision)
- Can we avoid it by using *Calculus 103*?



Figurnov, Mohamed, Mnih “Implicit Reparameterization Gradients”, 2018

Reminder: implicit differentiation

Implicit function $x^2 + y^2 = \phi, \quad x, y > 0 \quad \frac{\partial y}{\partial \phi} = ?$

Explicit differentiation

$$y = \sqrt{\phi - x^2}$$

$$\frac{\partial y}{\partial \phi} = \frac{1}{2\sqrt{\phi - x^2}}$$

Implicit differentiation

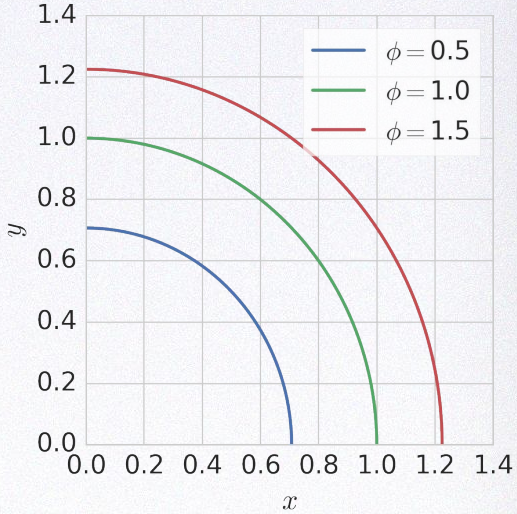
$$\frac{d}{d\phi}(x^2 + y^2) = \frac{d}{d\phi}\phi$$

$$2y\frac{\partial y}{\partial \phi} = 1$$

total derivative

$$\frac{\partial y}{\partial \phi} = \frac{1}{2y}$$

We don't need to compute x! 🤔



Implicit reparameterization gradients

Derivation

Standardization function: $\mathcal{S}_\phi(\boldsymbol{z}) = \boldsymbol{\varepsilon}$

$$\frac{d}{d\phi}\mathcal{S}_\phi(\boldsymbol{z}) = \frac{d}{d\phi}\boldsymbol{\varepsilon}$$

$$\frac{\partial \mathcal{S}_\phi(\boldsymbol{z})}{\partial \boldsymbol{z}}\frac{\partial \boldsymbol{z}}{\partial \phi} + \frac{\partial \mathcal{S}_\phi(\boldsymbol{z})}{\partial \phi} = \mathbf{0}$$

Solve for $\frac{\partial \boldsymbol{z}}{\partial \phi}$:

$$\underset{D\times 1}{\frac{\partial \boldsymbol{z}}{\partial \phi}} = -\underset{D\times D}{\left[\frac{\partial \mathcal{S}_\phi(\boldsymbol{z})}{\partial \boldsymbol{z}}\right]^{-1}}\underset{D\times 1}{\frac{\partial \mathcal{S}_\phi(\boldsymbol{z})}{\partial \phi}}$$

Figurnov, Mohamed, Mnih “Implicit Reparameterization Gradients”, 2018

Universal standardization function

for *continuous* distributions

- Univariate: CDF, $\mathcal{S}_\phi(z) = F(z|\phi) = u \sim \text{Uniform}(0, 1)$

$$\frac{\partial z}{\partial \phi} = -\frac{1}{q_\phi(z)} \frac{\partial F(z|\phi)}{\partial \phi}$$

1. Take the code that computes the CDF (e.g., using Taylor series)
2. Perform automatic differentiation

- Multivariate: distributional transform

$$\mathcal{S}_\phi(\boldsymbol{z}) = (F(z_1|\phi), F(z_2|z_1, \phi), \ldots, F(z_D|z_1, \ldots, z_{D-1}, \phi))$$
$$= \boldsymbol{u} \sim \prod_{d=1}^{D} \text{Uniform}(u_d|0, 1)$$

Figurnov, Mohamed, Mnih “Implicit Reparameterization Gradients”, 2018

Applications

- Multivariate mixture distribution
- Truncated distributions
- Gamma distribution
- Von Mises distribution

$$\frac{\partial \boldsymbol{z}}{\partial \phi} = -\left[\frac{\partial \mathcal{S}_\phi(\boldsymbol{z})}{\partial \boldsymbol{z}}\right]^{-1} \frac{\partial \mathcal{S}_\phi(\boldsymbol{z})}{\partial \phi}$$

- Inverse Gamma distribution
- Student's *t*-distribution
- Beta distribution
- Dirichlet distribution

Transformation of Gamma samples

Figurnov, Mohamed, Mnih "Implicit Reparameterization Gradients", 2018

Applications

Differentiable samples already implemented in TensorFlow

- ~~Multivariate mixture distribution~~
- Truncated Normal distribution — tfp.distributions.TruncatedNormal
- Gamma distribution — tf.distributions.Gamma
- Von Mises distribution — tfp.distributions.VonMises
- Inverse Gamma distribution — tfp.distributions.InverseGamma
- Student's *t*-distribution — tf.distributions.StudentT
- Beta distribution — tf.distributions.Beta
- Dirichlet distribution — tf.distributions.Dirichlet

Implicit reparameterization in TensorFlow / TensorFlow Probability!

Figurnov, Mohamed, Mnih "Implicit Reparameterization Gradients", 2018

Accuracy and speed of the gradient estimators

		Gamma		Von Mises	
Method	Precision	Mean abs. error	Time (s)	Mean abs. error	Time (s)
Automatic differentiation	`float32`	$\mathbf{2.3 \times 10^{-6}}$	$\mathbf{3.2 \times 10^{-7}}$	$\mathbf{1.9 \times 10^{-7}}$	$\mathbf{3.1 \times 10^{-7}}$
Finite difference		3.2×10^{-3}	3.7×10^{-7}	9.6×10^{-5}	3.8×10^{-7}
Automatic differentiation	`float64`	$\mathbf{5.4 \times 10^{-13}}$	$\mathbf{4.0 \times 10^{-7}}$	$\mathbf{1.3 \times 10^{-13}}$	$\mathbf{3.7 \times 10^{-7}}$
Finite difference		3.5×10^{-9}	6.7×10^{-7}	1.1×10^{-10}	5.9×10^{-7}
Knowles (2015)		6.5×10^{-3}	3.9×10^{-5}	–	–

Finite difference: $$\frac{\partial F(z|\phi)}{\partial \phi} \approx \frac{F(z|\phi+\delta) - F(z|\phi-\delta)}{2\delta}$$

Knowles (2015), approximate explicit reparameterization: $$\frac{\partial z}{\partial \phi} \approx \frac{F^{-1}\left(F(z|\phi)|\phi+\delta\right) - z}{\delta}$$

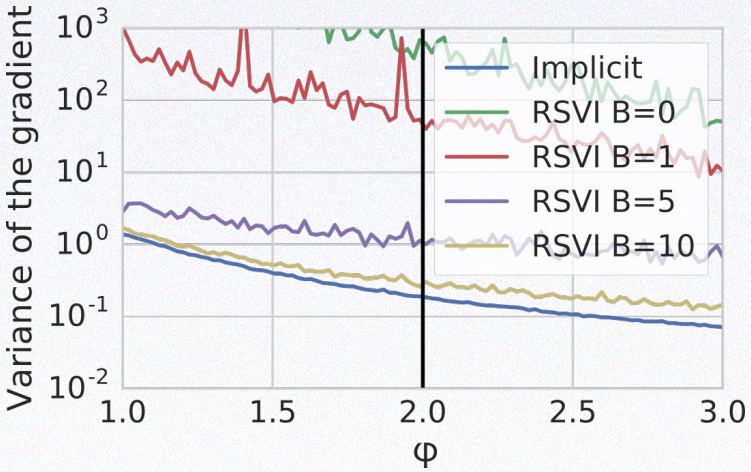
Knowles "Stochastic gradient variational Bayes for Gamma approximating distributions." arXiv, 2015

Related work

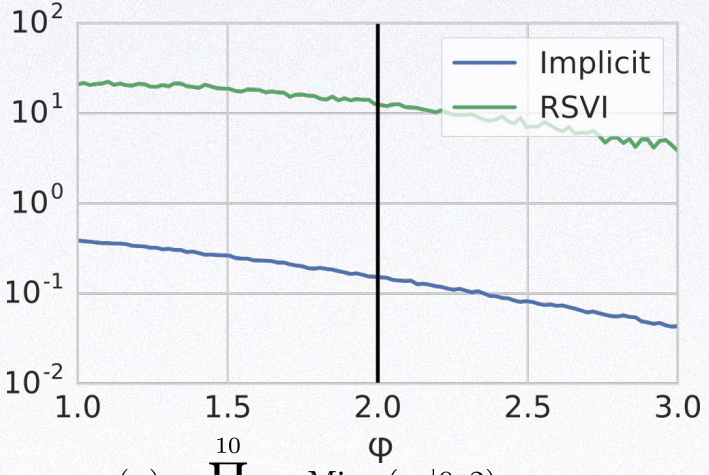
- Implicit (and explicit) reparameterization in operation research:
  - Suri, Zazanis “Perturbation analysis gives strongly consistent sensitivity estimates for the M/G/1 queue”. Management Science 34.1, 1988
- Implicit reparameterization in machine learning:
  - Salimans, Knowles “Fixed-form variational posterior approximation through stochastic linear regression”. Bayesian Analysis, 2013
  - Hoffman, Blei “Stochastic structured variational inference” AISTATS, 2015
  - Graves “Stochastic backpropagation through mixture density distributions” arXiv, 2016
  - Jankowiak, Obermeyer “Pathwise Derivatives Beyond the Reparameterization Trick” ICML, 2018 - concurrent work, implemented for Gamma in PyTorch
- Our contribution:
  - generalize to arbitrary standardization functions;
  - show the connection to explicit reparameterization gradients;
  - derive a simpler expression for the multivariate case, compared to Graves, 2016;
  - provide an efficient automatic differentiation method to compute intractable CDF derivatives.

Gradient of the cross-entropy $\frac{\partial}{\partial \phi} \mathbb{E}_{q_\phi(\boldsymbol{z})} [-\log p(\boldsymbol{z})]$

Implicit Reparameterization Gradients > Rejection Sampling Reparameterization Gradients



$$p(\boldsymbol{z}) = \text{Dirichlet}(\boldsymbol{z}|\alpha_1, \alpha_2, \ldots, \alpha_{100})$$
$$q_\phi(\boldsymbol{z}) = \text{Dirichlet}(\boldsymbol{z}|\phi, \alpha_2, \ldots, \alpha_{100})$$



$$p(\boldsymbol{z}) = \prod_{d=1}^{10} \text{vonMises}(z_d|0, 2)$$
$$q_\phi(\boldsymbol{z}) = \text{vonMises}(z_1|0, \phi) \prod_{d=2}^{10} \text{vonMises}(z_d|0, 2)$$

RSVI: Rejection Sampling Reparameterization

Latent Dirichlet Allocation

Topic model for documents represented as a bag-of-words

$$p(\boldsymbol{w}|\boldsymbol{\alpha}) = \int \left(\prod_{n=1}^{N} \text{Categorical}(w_n|\Phi\boldsymbol{\theta}) \right) \text{Dirichlet}(\boldsymbol{\theta}|\boldsymbol{\alpha}) d\boldsymbol{\theta}, \; \Phi \in \mathbb{R}^{\#\text{words} \times \#\text{topics}}, \; \sum_i \Phi_{ij} = 1$$

likelihood — linear decoder — prior — topic: distribution over words

Let's do black-box amortized inference! (Variational Autoencoder-like)

Approximate posterior / encoder: $q_\phi(\boldsymbol{\theta}|\boldsymbol{w}) = \text{Dirichlet}(\boldsymbol{\theta}|\,\text{NN}_\phi(\boldsymbol{w}))$

LDA: optimize ELBO using implicit reparameterization gradient

Logistic Normal LDA (LN-LDA): replace Dirichlet with Normal + Softmax in encoder and decoder

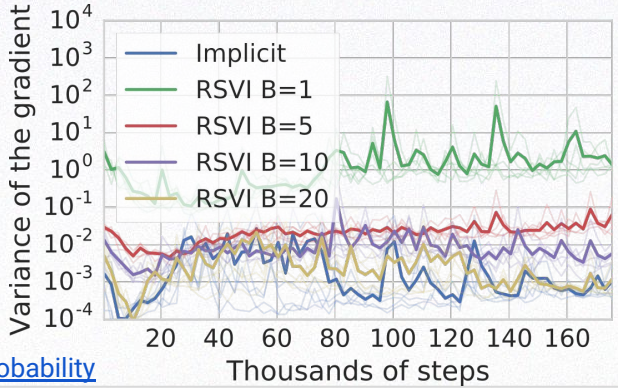
Blei, Ng, Jordan "Latent dirichlet allocation" JMLR, 2003
Srivastava, Sutton "Autoencoding variational inference for topic models.", ICLR 2017

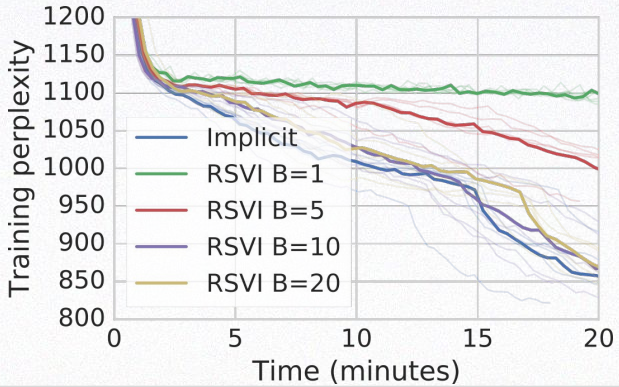
Latent Dirichlet Allocation

Test perplexity
(lower is better)

Model	Training method	20 Newsgroups	RCV1
LDA	Implicit reparameterization	876 ± 7	896 ± 6
	RSVI $B = 1$	1066 ± 7	1505 ± 33
	RSVI $B = 5$	968 ± 18	1075 ± 15
	RSVI $B = 10$	887 ± 10	953 ± 16
	RSVI $B = 20$	865 ± 11	907 ± 13
	SVI	964 ± 4	1330 ± 4
LN-LDA (Srivastava 2017)	Explicit reparameterization	875 ± 6	951 ± 10

Training curves for
20 newsgroups



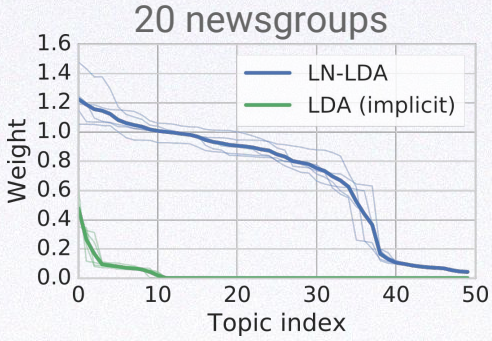


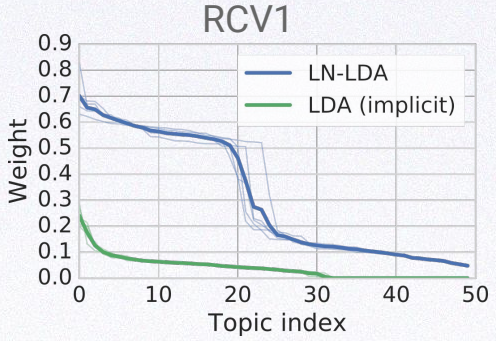
Code available in [TensorFlow Probability]

Latent Dirichlet Allocation

LDA produces sparse topics, unlike LN-LDA

Prior topic weights





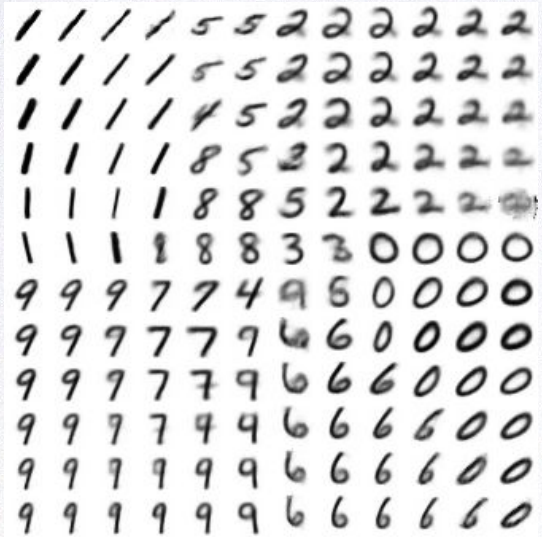
20 newsgroups topics

LN-LDA	LDA
$\alpha = 1.15$ write article get think go	$\alpha = 0.47$ write article get like one
$\alpha = 1.07$ write get think like article	$\alpha = 0.31$ write one people say think
$\alpha = 1.07$ write article get think like	$\alpha = 0.25$ please thanks post send know
$\alpha = 1.07$ write article get like know	$\alpha = 0.11$ use drive card problem system
$\alpha = 1.06$ write article think get like	$\alpha = 0.10$ go say people know get
$\alpha = 1.04$ write article get know think	$\alpha = 0.08$ use file key program system
$\alpha = 1.04$ write article get know like	$\alpha = 0.08$ gun government law state use
$\alpha = 1.02$ write article think get like	$\alpha = 0.08$ god christian jesus say people

Code available in [TensorFlow Probability]

Variational autoencoder

2D latent spaces for dynamically binarized MNIST



Normal posterior and prior
[-3, 3] x [-3, 3]

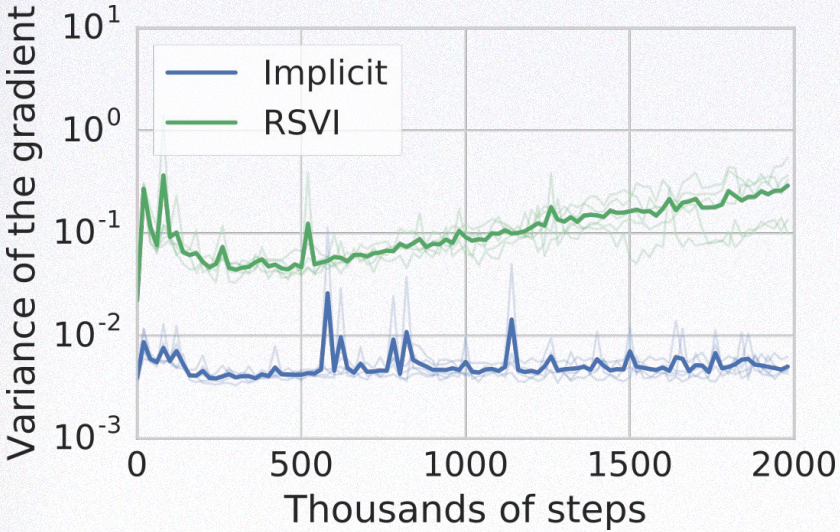
Beta posterior, Uniform prior
[0, 1] x [0, 1]

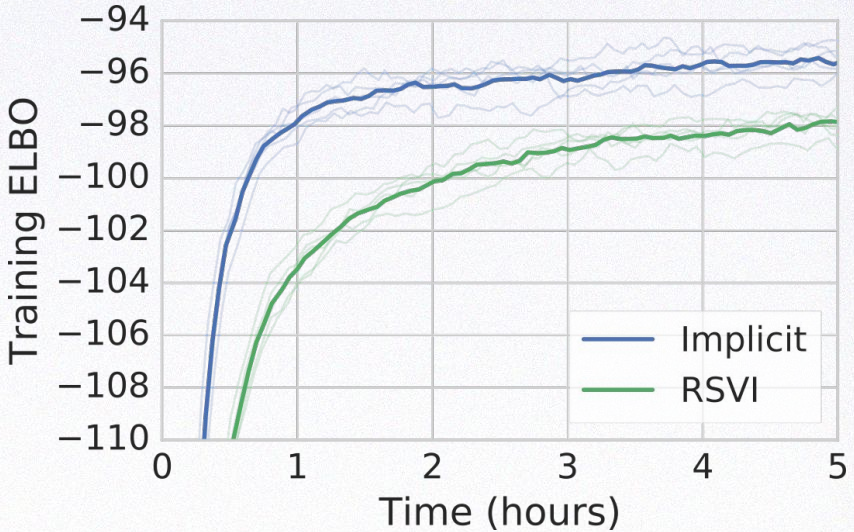
Von Mises posterior, Uniform prior
[-π, π] x [-π, π]

Variational autoencoder

Dynamically binarized MNIST

Training curves for for von Mises D=40





Outline

- Overview
  - Stochastic Gradient Estimation
  - REINFORCE
  - Reparameterization gradients
  - Comparison
- Expanding the applicability of reparameterization
  - Generalized Reparameterization Gradients
  - Implicit Reparameterization Gradients
- **Reducing the variance of the reparameterization gradient**
  - **Pathwise Derivatives for Multivariate Distributions**

Are reparameterization gradients unique?

- Univariate distribution (or factorized multivariate): **yes**

- Multivariate distribution: **no!**

Jankowiak, Karaletsos “Pathwise Derivatives for Multivariate Distributions”, 2018

Reparameterization for Multivariate Normal

Zero-mean for simplicity

Linear transformation of Multivariate Normal:

$$\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, I) \Rightarrow A\boldsymbol{\varepsilon} \sim \mathcal{N}(0, AA^T)$$

Covariance matrix $\Sigma(\phi) = L(\phi)L(\phi)^T$ $L(\phi)$ is lower-triangular; ϕ - e.g. some element of $L(\phi)$

Reparameterization for $q_\phi(\boldsymbol{z}) = \mathcal{N}(\mathbf{0}, \Sigma(\phi))$

$$\boldsymbol{z} = L\boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, I)$$

$$\frac{\partial \boldsymbol{z}}{\partial \phi} = L'\boldsymbol{\varepsilon} = L'L^{-1}\boldsymbol{z}$$

Alternative reparameterizations for Multivariate Normal

All linear inverse standardizations for Multivariate Normal have the form

$$\boldsymbol{z} = LQ\boldsymbol{\varepsilon}, \quad QQ^T = Q^TQ = I$$

Q is an orthogonal matrix

$$\boldsymbol{z} \sim \mathcal{N}(0, LQQ^TL^T) = \mathcal{N}(0, \Sigma)$$

What does $\frac{\partial \boldsymbol{z}}{\partial \phi}$ look like?

Jankowiak, Karaletsos “Pathwise Derivatives for Multivariate Distributions”, 2018 (different derivation)

Alternative reparameterizations for Multivariate Normal

$$\boldsymbol{z} = LQ\boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} = Q^{-1}L^{-1}\boldsymbol{z}$$

Property of orthogonal matrices that depend on a parameter:

$$Q'(\phi) = S(\phi)Q(\phi), \quad S(\phi) = -S(\phi)^T$$

skew-symmetric matrix

$$\frac{\partial \boldsymbol{z}}{\partial \phi} = (LQ)'\boldsymbol{\varepsilon} = (L'Q + LQ')\boldsymbol{\varepsilon} = (L'Q + LSQ)\boldsymbol{\varepsilon}$$

$$= (L'Q + LSQ)Q^{-1}L^{-1}\boldsymbol{z} = L'L^{-1}\boldsymbol{z} + LSL^{-1}\boldsymbol{z}$$

Jankowiak, Karaletsos “Pathwise Derivatives for Multivariate Distributions”, 2018 (different derivation)

The surprising control variate

that has almost the same form as the reparameterization gradient

$$\mathcal{L}'(\phi) = \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[\frac{\partial f(\boldsymbol{z})}{\partial \boldsymbol{z}} L' L^{-1} \boldsymbol{z}\right]$$

$$\mathcal{L}'(\phi) = \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[\frac{\partial f(\boldsymbol{z})}{\partial \boldsymbol{z}} (L' L^{-1} \boldsymbol{z} + L S L^{-1} \boldsymbol{z})\right]$$

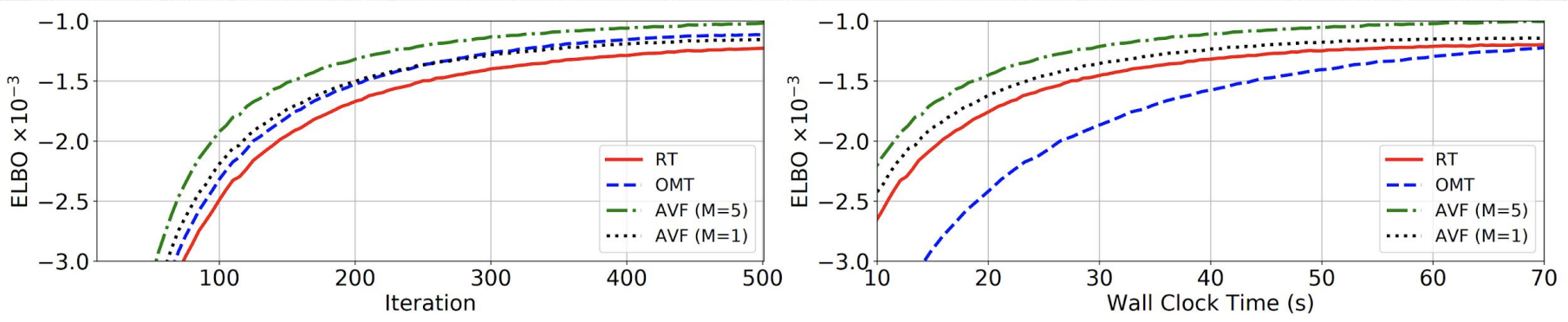
$$h(\boldsymbol{z}, \phi) = \frac{\partial f(\boldsymbol{z})}{\partial \boldsymbol{z}} L S L^{-1} \boldsymbol{z}, \quad \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[h(\boldsymbol{z}, \phi)\right] = 0$$

Optimize S to minimize the variance of the stochastic gradient (REBAR trick)

Jankowiak, Karaletsos “Pathwise Derivatives for Multivariate Distributions”, 2018 (different derivation)
Tucker, Mnih, Maddison, Lawson, Sohl-Dickstein “REBAR: Low-variance, unbiased gradient estimates for discrete latent variable models”, NIPS 2017

Results

Gaussian Process regression



- RT: standard reparameterization
- OMT: fixed Q that is optimal for $f(\mathbf{z}) = \mathbf{z}$
- AVF: RT + control variate; M is the flexibility of S

Jankowiak, Karaletsos “Pathwise Derivatives for Multivariate Distributions”, 2018

Conclusion

- Stochastic gradient estimation is a general and important problem
  - Reparameterization gradients and REINFORCE are two methods for this problem
  - Reparameterization gradients are lower variance for well-behaved cost functions
- Reparameterization can be applied to many continuous distributions
  - Generalized reparameterization gradients
  - Implicit reparameterization gradients
    - Have lower variance
    - Implemented in TensorFlow for many distributions
- Reparameterization for multivariate distributions is not unique
  - This leads to an interesting control variate for Multivariate Normal distribution

REINFORCE gradient estimator

Derivation

$$\mathcal{L}'(\phi) = \frac{\partial}{\partial \phi} \mathbb{E}_{q_\phi(\boldsymbol{z})} \left[f(\boldsymbol{z}) \right] = \frac{\partial}{\partial \phi} \int q_\phi(\boldsymbol{z}) f(\boldsymbol{z}) d\boldsymbol{z} = \int \frac{\partial q_\phi(\boldsymbol{z})}{\partial \phi} f(\boldsymbol{z}) d\boldsymbol{z}$$

$$= \left\{ \frac{\partial \log q_\phi(\boldsymbol{z})}{\partial \phi} = \frac{1}{q_\phi(\boldsymbol{z})} \frac{\partial q_\phi(\boldsymbol{z})}{\partial \phi} \right\} = \int q_\phi(\boldsymbol{z}) \frac{\partial \log q_\phi(\boldsymbol{z})}{\partial \phi} f(\boldsymbol{z}) d\boldsymbol{z}$$

log-derivative trick

$$= \mathbb{E}_{q_\phi(\boldsymbol{z})} \left[f(\boldsymbol{z}) \frac{\partial \log q_\phi(\boldsymbol{z})}{\partial \phi} \right]$$

Williams “Simple statistical gradient-following algorithms for connectionist reinforcement learning”. Machine Learning, 1992

Generalized Reparameterization Gradient

Derivation

$$\frac{\partial}{\partial\phi}\mathbb{E}_{q_\phi(\boldsymbol{z})}\left[f(\boldsymbol{z})\right] = \frac{\partial}{\partial\phi}\mathbb{E}_{q_\phi(\boldsymbol{z})}\left[\frac{r_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})}f(\boldsymbol{z})\right] = \frac{\partial}{\partial\phi}\mathbb{E}_{r_\phi(\boldsymbol{z})}\left[\frac{q_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})}f(\boldsymbol{z})\right]$$

$$= \mathbb{E}_{r_\phi(\boldsymbol{z})}\left[\frac{\partial}{\partial\phi}\left(\frac{q_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})}f(\boldsymbol{z})\right)\right] = \mathbb{E}_{r_\phi(\boldsymbol{z})}\left[\frac{\partial}{\partial\phi}\left(\frac{q_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})}\right)f(\boldsymbol{z}) + \frac{q_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})}\frac{\partial f(\boldsymbol{z})}{\partial\phi}\right]$$

$$= \mathbb{E}_{r_\phi(\boldsymbol{z})}\left[\frac{q_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})}\frac{\partial}{\partial\phi}\log\frac{q_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})}f(\boldsymbol{z}) + \frac{q_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})}\frac{\partial f(\boldsymbol{z})}{\partial\boldsymbol{z}}u_r(\boldsymbol{z},\phi)\right]$$

log-derivative trick

$$= \mathbb{E}_{q_\phi(\boldsymbol{z})}\left[f(\boldsymbol{z})\frac{\partial}{\partial\phi}\log\frac{q_\phi(\boldsymbol{z})}{r_\phi(\boldsymbol{z})} + \frac{\partial f(\boldsymbol{z})}{\partial\boldsymbol{z}}u_r(\boldsymbol{z},\phi)\right]$$

REINFORCE-like correction

reparameterization gradient for $r_\phi(\boldsymbol{z})$

Explicit vs. implicit reparameterization

	Explicit reparameterization	Implicit reparameterization
Forward pass	Sample $\boldsymbol{\varepsilon} \sim q(\boldsymbol{\varepsilon})$ $\boldsymbol{z} \leftarrow \mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon})$	Sample $\boldsymbol{z} \sim q_\phi(\boldsymbol{z})$
Backward pass	$\frac{\partial \boldsymbol{z}}{\partial \phi} \leftarrow \frac{\partial \mathcal{S}_\phi^{-1}(\boldsymbol{\varepsilon})}{\partial \phi}$ $\frac{\partial f(\boldsymbol{z})}{\partial \phi} \leftarrow \frac{\partial f(\boldsymbol{z})}{\partial \boldsymbol{z}} \frac{\partial \boldsymbol{z}}{\partial \phi}$	$\frac{\partial \boldsymbol{z}}{\partial \phi} \leftarrow -\left[\frac{\partial \mathcal{S}_\phi(\boldsymbol{z})}{\partial \boldsymbol{z}}\right]^{-1} \frac{\partial \mathcal{S}_\phi(\boldsymbol{z})}{\partial \phi}$ $\frac{\partial f(\boldsymbol{z})}{\partial \phi} \leftarrow \frac{\partial f(\boldsymbol{z})}{\partial \boldsymbol{z}} \frac{\partial \boldsymbol{z}}{\partial \phi}$

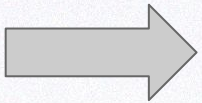
Figurnov, Mohamed, Mnih “Implicit Reparameterization Gradients”, 2018

Computing the derivative of the CDF

Forward-mode “automatic” differentiation of C++ code

Computes CDF of Gamma distribution

```
Scalar r = a;
...
for (int i = 0; i < 200; i++) {
   r += 1;
   Scalar term = x / r;
   c *= term;
   ans += c;
   if (c <= machep * ans) break;
}
```



Computes CDF and **∂CDF / ∂a**

```
Scalar r = a;
...
for (int i = 0; i < 200; i++) {
   r += 1;
   Scalar term = x / r;
   Scalar dterm_da = -x / (r * r);
   dc_da = term * dc_da + dterm_da * c;
   dans_da += dc_da;
   c *= term;
   ans += c;
   if (abs(dc_da) <= machep * abs(dans_da)) break;
}
```

Code based on Eigen library (MPL-licensed)

Are reparameterization gradients unique?

- Univariate distribution (or factorized multivariate): **yes**
  - Transform any univariate standardization function into the CDF with a function independent of ϕ
  - $\frac{\partial z}{\partial \phi}$ does not change under such transformations: the derivatives of the transformation cancel out

- Multivariate distribution: **no!**
  - We can rotate or permute $\mathbf{z}$ before standardizing

Jankowiak, Karaletsos “Pathwise Derivatives for Multivariate Distributions”, 2018

Alternative reparameterizations for Multivariate Normal

$$A\boldsymbol{\varepsilon} = \mathcal{N}(0, AA^T) = \mathcal{N}(0, \Sigma) \Leftrightarrow AA^T = \Sigma$$

Easy to show that $A = LQ, \quad QQ^T = Q^TQ = I$

Property of orthogonal matrices that depend on a parameter:

$$Q'(\phi) = S(\phi)Q(\phi), \quad S(\phi) = -S(\phi)^T$$ skew-symmetric matrix

Example: $$Q(\phi) = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix}$$

$$Q'(\phi) = \begin{pmatrix} -\sin\phi & -\cos\phi \\ \cos\phi & -\sin\phi \end{pmatrix} = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} Q(\phi)$$