Reparameterization Gradients through Acceptance-Rejection Sampling Algorithms

Christian A. Naesseth[†‡] **Francisco J. R. Ruiz**[‡§] **Scott W. Linderman**[‡] **David M. Blei**[‡]
[†]Linköping University [‡]Columbia University [§]University of Cambridge

Abstract

Variational inference using the reparameterization trick has enabled large-scale approximate Bayesian inference in complex probabilistic models, leveraging stochastic optimization to sidestep intractable expectations. The reparameterization trick is applicable when we can simulate a random variable by applying a differentiable deterministic function on an auxiliary random variable whose distribution is fixed. For many distributions of interest (such as the gamma or Dirichlet), simulation of random variables relies on acceptance-rejection sampling. The discontinuity introduced by the accept–reject step means that standard reparameterization tricks are not applicable. We propose a new method that lets us leverage reparameterization gradients even when variables are outputs of a acceptance-rejection sampling algorithm. Our approach enables reparameterization on a larger class of variational distributions. In several studies of real and synthetic data, we show that the variance of the estimator of the gradient is significantly lower than other state-of-the-art methods. This leads to faster convergence of stochastic gradient variational inference.

1 Introduction

Variational inference [Hinton and van Camp, 1993, Waterhouse et al., 1996, Jordan et al., 1999] underlies many recent advances in large scale probabilistic modeling. It has enabled sophisticated modeling of complex domains such as images [Kingma and Welling, 2014] and text [Hoffman et al., 2013]. By definition, the success of variational approaches depends on our ability to (i) formulate a flexible parametric family of distributions; and (ii) optimize the parameters to find the member of this family that most closely approximates the true posterior. These two criteria are at odds—the more flexible the family, the more challenging the optimization problem. In this paper, we present a novel method that enables more efficient optimization for a large class of variational distributions, namely, for distributions that we can efficiently simulate by acceptance-rejection sampling, or rejection sampling for short.

For complex models, the variational parameters can be optimized by stochastic gradient ascent on the evidence lower bound (ELBO), a lower bound on the marginal likelihood of the data. There are two primary means of estimating the gradient of the ELBO: the score function estimator [Paisley et al., 2012, Ranganath et al., 2014, Mnih and Gregor, 2014] and the reparameterization trick [Kingma and Welling, 2014, Rezende et al., 2014, Price, 1958, Bonnet, 1964], both of which rely on Monte Carlo sampling. While the reparameterization trick often yields lower variance estimates and therefore leads to more efficient optimization, this approach has been limited in scope to a few variational families (typically Gaussians). Indeed, some lines of research have already tried to address this limitation [Knowles, 2015, Ruiz et al., 2016].

There are two requirements to apply the reparameterization trick. The first is that the random variable can be obtained through a transformation of a simple random variable, such as a uniform or standard normal; the second is that the transformation be differentiable. In this paper, we observe that all random variables we simulate on our computers are ultimately transformations of uniforms, often followed by accept-reject steps. So if the transformations are differentiable then we can use these existing simulation algorithms to expand the scope of the reparameterization trick.

Thus, we show how to use existing rejection samplers

Proceedings of the 20^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2017, Fort Lauderdale, Florida, USA. JMLR: W&CP volume 54. Copyright 2017 by the author(s).

to develop stochastic gradients of variational parameters. In short, each rejection sampler uses a highly-tuned transformation that is well-suited for its distribution. We can construct new reparameterization gradients by "removing the lid" from these black boxes, applying 65+ years of research on transformations [von Neumann, 1951, Devroye, 1986] to variational inference. We demonstrate that this broadens the scope of variational models amenable to efficient inference and provides lower-variance estimates of the gradient compared to state-of-the-art approaches.

We first review variational inference, with a focus on stochastic gradient methods. We then present our key contribution, rejection sampling variational inference (RSVI), showing how to use efficient rejection samplers to produce low-variance stochastic gradients of the variational objective. We study two concrete examples, analyzing rejection samplers for the gamma and Dirichlet to produce new reparameterization gradients for their corresponding variational factors. Finally, we analyze two datasets with a deep exponential family (DEF) [Ranganath et al., 2015], comparing RSVI to the state of the art. We found that RSVI achieves a significant reduction in variance and faster convergence of the ELBO. Code for all experiments is provided at `github.com/blei-lab/ars-reparameterization`.

2 Variational Inference

Let $p(x, z)$ be a probabilistic model, i.e., a joint probability distribution of *data* x and *latent* (unobserved) variables z. In Bayesian inference, we are interested in the posterior distribution $p(z|x) = \frac{p(x,z)}{p(x)}$. For most models, the posterior distribution is analytically intractable and we have to use an approximation, such as Monte Carlo methods or variational inference. In this paper, we focus on variational inference.

In variational inference, we approximate the posterior with a *variational family* of distributions $q(z\,;\theta)$, parameterized by θ. Typically, we choose the *variational parameters* θ that minimize the Kullback-Leibler (KL) divergence between $q(z\,;\theta)$ and $p(z|x)$. This minimization is equivalent to maximizing the ELBO [Jordan et al., 1999], defined as

$$\begin{aligned} \mathcal{L}(\theta) &= \mathbb{E}_{q(z\,;\theta)}\left[f(z)\right] + \mathbb{H}[q(z\,;\theta)], \\ f(z) &:= \log p(x, z), \\ \mathbb{H}[q(z\,;\theta)] &:= \mathbb{E}_{q(z\,;\theta)}[-\log q(z\,;\theta)]. \end{aligned} \tag{1}$$

When the model and variational family satisfy conjugacy requirements, we can use coordinate ascent to find a local optimum of the ELBO [Blei et al., 2016]. If the conjugacy requirements are not satisfied, a common approach is to build a Monte Carlo estimator of the gradient of the ELBO [Paisley et al., 2012, Ranganath et al., 2014, Mnih and Gregor, 2014, Salimans and Knowles, 2013, Kingma and Welling, 2014]. This results in a stochastic optimization procedure, where different Monte Carlo estimators of the gradient amount to different algorithms. We review below two common estimators: the score function estimator and the reparameterization trick.[1]

Score function estimator. The score function estimator, also known as the log-derivative trick or REINFORCE [Williams, 1992, Glynn, 1990], is a general way to estimate the gradient of the ELBO [Paisley et al., 2012, Ranganath et al., 2014, Mnih and Gregor, 2014]. The score function estimator expresses the gradient as an expectation with respect to $q(z\,;\theta)$:

$$\nabla_\theta \mathcal{L}(\theta) = \mathbb{E}_{q(z\,;\theta)}[f(z)\nabla_\theta \log q(z\,;\theta)] + \nabla_\theta \mathbb{H}[q(z\,;\theta)].$$

We then form Monte Carlo estimates by approximating the expectation with independent samples from the variational distribution. Though it is very general, the score function estimator typically suffers from high variance. In practice we also need to apply variance reduction techniques such as Rao-Blackwellization [Casella and Robert, 1996] and control variates [Robert and Casella, 2004].

Reparameterization trick. The reparameterization trick [Salimans and Knowles, 2013, Kingma and Welling, 2014, Price, 1958, Bonnet, 1964] results in a lower variance estimator compared to the score function, but it is not as generally applicable. It requires that: (i) the latent variables z are continuous; and (ii) we can simulate from $q(z\,;\theta)$ as follows,

$$z = h(\varepsilon, \theta), \qquad \text{with } \varepsilon \sim s(\varepsilon). \tag{2}$$

Here, $s(\varepsilon)$ is a distribution that does not depend on the variational parameters; it is typically a standard normal or a standard uniform. Further, $h(\varepsilon, \theta)$ must be differentiable with respect to θ. In statistics, this is known as a non-central parameterization and has been shown to be helpful in, e.g., Markov chain Monte Carlo methods [Papaspiliopoulos et al., 2003].

Using (2), we can move the derivative inside the expectation and rewrite the gradient of the ELBO as

$$\nabla_\theta \mathcal{L}(\theta) = \mathbb{E}_{s(\varepsilon)}\left[\nabla_z f(h(\varepsilon, \theta))\nabla_\theta h(\varepsilon, \theta)\right] + \nabla_\theta \mathbb{H}[q(z\,;\theta)].$$

Empirically, the reparameterization trick has been shown to be beneficial over direct Monte Carlo es-

[1] In this paper, we assume for simplicity that the gradient of the entropy $\nabla_\theta \mathbb{H}[q(z\,;\theta)]$ is available analytically. The method that we propose in Section 3 can be easily extended to handle non-analytical entropy terms. Indeed, the resulting estimator of the gradient may have lower variance when the analytic gradient of the entropy is replaced by its Monte Carlo estimate. Here we do not explore that.

timation of the gradient using the score fuction estimator [Salimans and Knowles, 2013, Kingma and Welling, 2014, Titsias and Lázaro-Gredilla, 2014, Fan et al., 2015]. Unfortunately, many distributions commonly used in variational inference, such as gamma or Dirichlet, are not amenable to standard reparameterization because samples are generated using a rejection sampler [von Neumann, 1951, Robert and Casella, 2004], introducing discontinuities to the mapping. We next show that taking a novel view of the acceptance-rejection sampler lets us perform exact reparameterization.

Algorithm 1 Reparameterized Rejection Sampling

```
Input: target $q(z\,;\theta)$, proposal $r(z\,;\theta)$, and constant
    $M_\theta$, with $q(z\,;\theta) \leq M_\theta r(z\,;\theta)$
Output: $\varepsilon$ such that $h(\varepsilon,\theta) \sim q(z\,;\theta)$
$i \leftarrow 0$
repeat
    $i \leftarrow i+1$
    Propose $\varepsilon_i \sim s(\varepsilon)$
    Simulate $u_i \sim \mathcal{U}[0,1]$
until $u_i < \frac{q(h(\varepsilon_i,\theta)\,;\theta)}{M_\theta r(h(\varepsilon_i,\theta)\,;\theta)}$
return $\varepsilon_i$
```

3 Reparameterizing the Acceptance-Rejection Sampler

The basic idea behind reparameterization is to rewrite simulation from a complex distribution as a deterministic mapping of its parameters and a set of simpler random variables. We can view the rejection sampler as a complicated deterministic mapping of a (random) number of simple random variables such as uniforms and normals. This makes it tempting to take the standard reparameterization approach when we consider random variables generated by rejection samplers. However, this mapping is in general not continuous, and thus moving the derivative inside the expectation and using direct automatic differentiation would not necessarily give the correct answer.

Our insight is that we can overcome this problem by instead considering only the marginal over the accepted sample, analytically integrating out the accept-reject variable. Thus, the mapping comes from the proposal step. This is continuous under mild assumptions, enabling us to greatly extend the class of variational families amenable to reparameterization.

We first review rejection sampling and present the reparameterized rejection sampler. Next we show how to use it to calculate low-variance gradients of the ELBO. Finally, we present the complete stochastic optimization for variational inference, RSVI.

3.1 Reparameterized Rejection Sampling

Acceptance-Rejection sampling is a powerful way of simulating random variables from complex distributions whose inverse cumulative distribution functions are not available or are too expensive to evaluate [Devroye, 1986, Robert and Casella, 2004]. We consider an alternative view of rejection sampling in which we explicitly make use of the reparameterization trick. This view of the rejection sampler enables our variational inference algorithm in Section 3.2.

To generate samples from a distribution $q(z\,;\theta)$ using rejection sampling, we first sample from a *proposal distribution* $r(z\,;\theta)$ such that $q(z\,;\theta) \leq M_\theta r(z\,;\theta)$ for some $M_\theta < \infty$. In our version of the rejection sampler, we assume that the proposal distribution is reparameterizable, i.e., that generating $z \sim r(z\,;\theta)$ is equivalent to generating $\varepsilon \sim s(\varepsilon)$ (where $s(\varepsilon)$ does not depend on θ) and then setting $z = h(\varepsilon,\theta)$ for a differentiable function $h(\varepsilon,\theta)$. We then accept the sample with probability $\min\left\{1, \frac{q(h(\varepsilon,\theta)\,;\theta)}{M_\theta r(h(\varepsilon,\theta)\,;\theta)}\right\}$; otherwise, we reject the sample and repeat the process. We illustrate this in Figure 1 and provide a summary of the method in Algorithm 1, where we consider the output to be the (accepted) variable ε, instead of z.

The ability to simulate from $r(z\,;\theta)$ by a reparameterization through a differentiable $h(\varepsilon,\theta)$ is not needed for the rejection sampler to be valid. However, this is indeed the case for the rejection sampler of many common distributions.

3.2 The Reparameterized Rejection Sampler in Variational Inference

We now use reparameterized rejection sampling to develop a novel Monte Carlo estimator of the gradient of the ELBO. We first rewrite the ELBO in (1) as an expectation in terms of the transformed variable ε,

$$\begin{aligned}\mathcal{L}(\theta) &= \mathbb{E}_{q(z\,;\theta)}\left[f(z)\right] + \mathbb{H}[q(z\,;\theta)] \\ &= \mathbb{E}_{\pi(\varepsilon\,;\theta)}\left[f\left(h(\varepsilon,\theta)\right)\right] + \mathbb{H}[q(z\,;\theta)].\end{aligned} \tag{3}$$

In this expectation, $\pi(\varepsilon\,;\theta)$ is the distribution of the *accepted sample* ε in Algorithm 1. We construct it by marginalizing over the auxiliary uniform variable u,

$$\begin{aligned}\pi(\varepsilon\,;\theta) &= \int \pi(\varepsilon, u\,;\theta)\mathrm{d}u \\ &= \int M_\theta s(\varepsilon)\mathbb{1}\left[0 < u < \frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{M_\theta r\left(h(\varepsilon,\theta)\,;\theta\right)}\right]\mathrm{d}u \\ &= s(\varepsilon)\frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)},\end{aligned} \tag{4}$$

where $\mathbb{1}[x \in A]$ is the indicator function, and recall that M_θ is a constant used in the rejection sam-

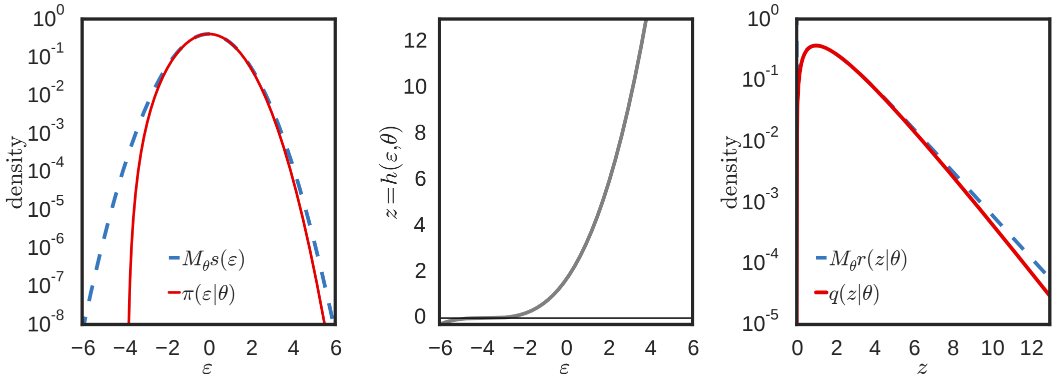

Figure 1: Example of a reparameterized rejection sampler for $q(z\,;\theta) = \text{Gamma}(\theta, 1)$, shown here with $\theta = 2$. We use the rejection sampling algorithm of Marsaglia and Tsang [2000], which is based on a nonlinear transformation $h(\varepsilon, \theta)$ of a standard normal $\varepsilon \sim \mathcal{N}(0, 1)$ (c.f. Eq. 10), and has acceptance probability of 0.98 for $\theta = 2$. The marginal density of the accepted value of ε (integrating out the acceptance variables, $u_{1:i}$) is given by $\pi(\varepsilon\,;\theta)$. We compute unbiased estimates of the gradient of the ELBO (6) via Monte Carlo, using Algorithm 1 to rejection sample $\varepsilon \sim \pi(\varepsilon\,;\theta)$. By reparameterizing in terms of ε, we obtain a low-variance estimator of the gradient for challenging variational distributions.

pler. This can be seen by the algorithmic definition of the rejection sampler, where we propose values $\varepsilon \sim s(\varepsilon)$ and $u \sim \mathcal{U}[0, 1]$ until acceptance, i.e., until $u < \frac{q(h(\varepsilon,\theta)\,;\theta)}{M_\theta r(h(\varepsilon,\theta)\,;\theta)}$. Eq. 3 follows intuitively, but we formalize it in Proposition 1.

Proposition 1. *Let f be any measurable function, and $\varepsilon \sim \pi(\varepsilon\,;\theta)$, defined by* (4) *(and implicitly by Algorithm 1). Then*

$$\mathbb{E}_{\pi(\varepsilon\,;\theta)}\left[f\left(h(\varepsilon, \theta)\right)\right] = \int f(z) q(z\,;\theta) dz.$$

Proof. Using the definition of $\pi(\varepsilon\,;\theta)$,

$$\mathbb{E}_{\pi(\varepsilon\,;\theta)}\left[f\left(h(\varepsilon, \theta)\right)\right] = \int f\left(h(\varepsilon, \theta)\right) s(\varepsilon) \frac{q\left(h(\varepsilon, \theta)\,;\theta\right)}{r\left(h(\varepsilon, \theta)\,;\theta\right)} d\varepsilon$$
$$= \int f(z) r(z\,;\theta) \frac{q(z\,;\theta)}{r(z\,;\theta)} dz = \int f(z) q(z\,;\theta) dz,$$

where the second to last equality follows because $h(\varepsilon, \theta), \varepsilon \sim s(\varepsilon)$ is a reparameterization of $r(z\,;\theta)$. □

We can now compute the gradient of $\mathbb{E}_{q(z\,;\theta)}[f(z)]$ based on Eq. 3,

$$\begin{aligned}
\nabla_\theta \mathbb{E}_{q(z\,;\theta)}[f(z)] &= \nabla_\theta \mathbb{E}_{\pi(\varepsilon\,;\theta)}[f(h(\varepsilon, \theta))] \\
&= \underbrace{\mathbb{E}_{\pi(\varepsilon\,;\theta)}[\nabla_\theta f(h(\varepsilon, \theta))]}_{=:g_{\text{rep}}} + \\
&\quad + \underbrace{\mathbb{E}_{\pi(\varepsilon\,;\theta)}\left[f(h(\varepsilon, \theta)) \nabla_\theta \log \frac{q(h(\varepsilon, \theta)\,;\theta)}{r(h(\varepsilon, \theta)\,;\theta)}\right]}_{=:g_{\text{cor}}},
\end{aligned} \tag{5}$$

where we have used the log-derivative trick and rewritten the integrals as expectations with respect to $\pi(\varepsilon\,;\theta)$ (see the supplement for all details.) We define g_{rep} as the reparameterization term, which takes advantage of gradients with respect to the model and its latent variables; we define g_{cor} as a correction term that accounts for *not* using $r(z\,;\theta) \equiv q(z\,;\theta)$.

Using (5), the gradient of the ELBO in (1) can be written as

$$\nabla_\theta \mathcal{L}(\theta) = g_{\text{rep}} + g_{\text{cor}} + \nabla_\theta \mathbb{H}[q(z\,;\theta)], \tag{6}$$

and thus we can build an unbiased one-sample Monte Carlo estimator $\hat{g} \approx \nabla_\theta \mathcal{L}(\theta)$ as

$$\begin{aligned}
\hat{g} &:= \hat{g}_{\text{rep}} + \hat{g}_{\text{cor}} + \nabla_\theta \mathbb{H}[q(z\,;\theta)], \\
\hat{g}_{\text{rep}} &= \nabla_z f(z)\big|_{z=h(\varepsilon,\theta)} \nabla_\theta h(\varepsilon, \theta) \\
\hat{g}_{\text{cor}} &= f(h(\varepsilon, \theta)) \nabla_\theta \log \frac{q(h(\varepsilon, \theta)\,;\theta)}{r(h(\varepsilon, \theta)\,;\theta)},
\end{aligned} \tag{7}$$

where ε is a sample generated using Algorithm 1. Of course, one could generate more samples of ε and average, but we have found a single sample to suffice in practice.

Note if $h(\varepsilon, \theta)$ is invertible in ε then we can simplify the evaluation of the gradient of the log-ratio in g_{cor},

$$\begin{aligned}
&\nabla_\theta \log \frac{q(h(\varepsilon, \theta)\,;\theta)}{r(h(\varepsilon, \theta)\,;\theta)} = \\
&\qquad \nabla_\theta \log q(h(\varepsilon, \theta)\,;\theta) + \nabla_\theta \log \left|\frac{dh}{d\varepsilon}(\varepsilon, \theta)\right|.
\end{aligned} \tag{8}$$

See the supplementary material for details.

Alternatively, we could rewrite the gradient as an expectation with respect to $s(\varepsilon)$ (this is an intermediate step in the derivation shown in the supplement),

$$\nabla_\theta \mathbb{E}_{q(z\,;\theta)}[f(z)] = \mathbb{E}_{s(\varepsilon)}\left[\frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)}\nabla_\theta f\left(h(\varepsilon,\theta)\right)\right] + $$
$$+\,\mathbb{E}_{s(\varepsilon)}\left[\frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)} f\left(h(\varepsilon,\theta)\right)\nabla_\theta \log\frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)}\right],$$

and build an importance sampling-based Monte Carlo estimator, in which the importance weights would be $q\left(h(\varepsilon,\theta)\,;\theta\right)/r\left(h(\varepsilon,\theta)\,;\theta\right)$. However, we would expect this approach to be beneficial for low-dimensional problems only, since for high-dimensional z the variance of the importance weights would be too high.

Algorithm 2 Rejection Sampling Variational Inference

```
Input: Data $x$, model $p(x,z)$, variational family $q(z\,;\theta)$
Output: Variational parameters $\theta^*$
1: repeat
2:     Run Algorithm 1 for $\theta^n$ to obtain a sample $\varepsilon$
3:     Estimate the gradient $\hat{g}^n$ at $\theta = \theta^n$ (Eq. 7)
4:     Calculate the stepsize $\rho^n$ (Eq. 9)
5:     Update $\theta^{n+1} = \theta^n + \rho^n \hat{g}^n$
6: until convergence
```

3.3 Full Algorithm

We now describe the full variational algorithm based on reparameterizing the rejection sampler. In Section 5 we give concrete examples of how to reparameterize common variational families.

We make use of Eq. 6 to obtain a Monte Carlo estimator of the gradient of the ELBO. We use this estimate to take stochastic gradient steps. We use the step-size sequence ρ^n proposed by Kucukelbir et al. [2016] (also used by Ruiz et al. [2016]), which combines RMSPROP [Tieleman and Hinton, 2012] and Adagrad [Duchi et al., 2011]. It is

$$\begin{aligned}\rho^n &= \eta \cdot n^{-1/2+\delta}\cdot\left(1+\sqrt{s^n}\right)^{-1},\\ s^n &= t\left(\hat{g}^n\right)^2 + (1-t)s^{n-1},\end{aligned}\tag{9}$$

where n is the iteration number. We set $\delta = 10^{-16}$ and $t = 0.1$, and we try different values for η. (When θ is a vector, the operations above are element-wise.)

We summarize the full method in Algorithm 2. We refer to our method as RSVI.

4 Related Work

The reparameterization trick has also been used in automatic differentiation variational inference (ADVI) [Kucukelbir et al., 2015, 2016]. ADVI applies a transformation to the random variables such that their support is on the reals and then places a Gaussian variational posterior approximation over the transformed variable ε. In this way, ADVI allows for standard reparameterization, but it cannot fit gamma or Dirichlet variational posteriors, for example. Thus, ADVI struggles to approximate probability densities with singularities, as noted by Ruiz et al. [2016]. In contrast, our approach allows us to apply the reparameterization trick on a wider class of variational distributions, which may be more appropriate when the exact posterior exhibits sparsity.

In the literature, we can find other lines of research that focus on extending the reparameterization gradient to other distributions. For the gamma distribution, Knowles [2015] proposed a method based on approximations of the inverse cumulative density function; however, this approach is limited only to the gamma distribution and it involves expensive computations. For general expectations, Schulman et al. [2015] expressed the gradient as a sum of a reparameterization term and a correction term to automatically estimate the gradient in the context of stochastic computation graphs. However, it is not possible to directly apply it to variational inference with acceptance-rejection sampling. This is due to discontinuities in the accept–reject step and the fact that a rejection sampler produces a *random number* of random variables. Recently, another line of work has focused on applying reparameterization to discrete latent variable models [Maddison et al., 2017, Jang et al., 2017] through a continuous relaxation of the discrete space.

The generalized reparameterization (G-REP) method [Ruiz et al., 2016] exploits the decomposition of the gradient as $g_{\text{rep}} + g_{\text{cor}}$ by applying a transformation based on standardization of the sufficient statistics of z. Our approach differs from G-REP: instead of searching for a transformation of z that makes the distribution of ε weakly dependent on the variational parameters (namely, standardization), we do the opposite by choosing a transformation of a simple random variable ε such that the distribution of $z = h(\varepsilon,\theta)$ is *almost* equal to $q(z\,;\theta)$. For that, we reuse the transformations typically used in rejection sampling. Rather than having to derive a new transformation for each variational distribution, we leverage decades of research on transformations in the rejection sampling literature [Devroye, 1986]. In rejection sampling, these transformations (and the distributions of ε) are chosen so that they have high acceptance probability, which means we should expect to obtain $g_{\text{cor}} \approx 0$ with RSVI. In Sections 5 and 6 we compare RSVI with G-REP and show that it exhibits significantly lower variance, thus leading to faster convergence of the inference algorithm.

Finally, another line of research in non-conjugate variational inference aims at developing more expressive variational families [Salimans et al., 2015, Tran et al., 2016, Maaløe et al., 2016, Ranganath et al., 2016]. RSVI can extend the reparameterization trick to these methods as well, whenever rejection sampling is used to generate the random variables.

5 Examples of Acceptance-Rejection Reparameterization

As two examples, we study rejection sampling and reparameterization of two well-known distributions: the gamma and Dirichlet. These have been widely used as variational families for approximate Bayesian inference. We emphasize that RSVI is not limited to these two cases, it applies to any variational family $q(z\,;\theta)$ for which a reparameterizable rejection sampler exists. We provide other examples in the supplement.

5.1 Gamma Distribution

One of the most widely used rejection sampler is for the gamma distribution. Indeed, the gamma distribution is also used in practice to generate e.g. beta, Dirichlet, and Student's t-distributed random variables. The gamma distribution, $\text{Gamma}(\alpha, \beta)$, is defined by its shape α and rate β.

For $\text{Gamma}(\alpha, 1)$ with $\alpha \geq 1$, Marsaglia and Tsang [2000] developed an efficient rejection sampler. It uses a truncated version of the following reparameterization

$$z = h_{\text{Gamma}}(\varepsilon, \alpha) := \left(\alpha - \frac{1}{3}\right)\left(1 + \frac{\varepsilon}{\sqrt{9\alpha - 3}}\right)^3, \tag{10}$$
$$\varepsilon \sim s(\varepsilon) := \mathcal{N}(0, 1).$$

When $\beta \neq 1$, we divide z by the rate β and obtain a sample distributed as $\text{Gamma}(\alpha, \beta)$. The acceptance probability is very high: it exceeds 0.95 and 0.98 for $\alpha = 1$ and $\alpha = 2$, respectively. In fact, as $\alpha \to \infty$ we have that $\pi(\varepsilon\,;\theta) \to s(\varepsilon)$, which means that the acceptance probability approaches 1. Figure 1 illustrates the involved functions and distributions for shape $\alpha = 2$.

For $\alpha < 1$, we observe that $z = u^{1/\alpha}\tilde{z}$ is distributed as $\text{Gamma}(\alpha, \beta)$ for $\tilde{z} \sim \text{Gamma}(\alpha{+}1, \beta)$ and $u \sim \mathcal{U}[0, 1]$ [Stuart, 1962, Devroye, 1986], and apply the rejection sampler above for $\tilde{z}$.

We now study the quality of the transformation in (10) for different values of the shape parameter α. Since $\pi(\varepsilon\,;\theta) \to s(\varepsilon)$ as $\alpha \to \infty$, we should expect the correction term g_{cor} to decrease with α. We show that in Figure 2, where we plot the log-ratio (8) from the correction term as a function of ε for four values of α. We additionally show in Figure 3 that the distribution $\pi(\varepsilon\,;\theta)$ converges to $s(\varepsilon)$ (a standard normal) as α increases. For large α, $\pi(\varepsilon\,;\theta) \approx s(\varepsilon)$ and the acceptance probability of the rejection sampler approaches 1, which makes the correction term negligible. In Figure 3, we also show that $\pi(\varepsilon\,;\theta)$ converges faster to a standard normal than the standardization procedure used in G-REP. We exploit this property—that performance improves with α—to artificially increase the shape for any gamma distribution. We now explain this trick, which we call *shape augmentation*.

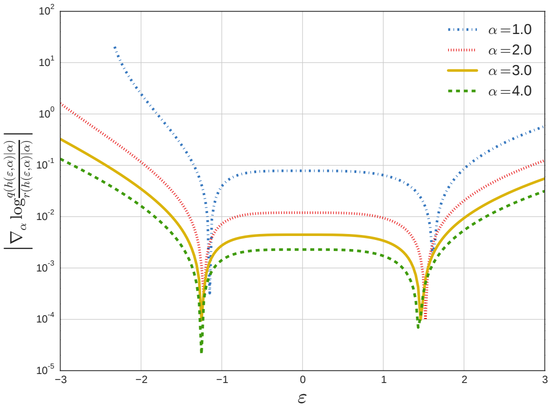

Figure 2: The correction term of RSVI, and as a result the gradient variance, decreases with increasing shape α. We plot absolute value of the gradient of the log-ratio between the target (gamma) and proposal distributions as a function of ε.

Shape augmentation. Here we show how to exploit the fact that the rejection sampler improves for increasing shape α. We make repeated use of the trick above, using uniform variables, to control the value of α that goes into the rejection sampler. That is, to compute the ELBO for a $\text{Gamma}(\alpha, 1)$ distribution, we can first express the random variable as $z = \tilde{z}\prod_{i=1}^{B} u_i^{\frac{1}{\alpha+i-1}}$ (for some positive integer B), $\tilde{z} \sim \text{Gamma}(\alpha + B, 1)$ and $u_i \overset{\text{i.i.d.}}{\sim} \mathcal{U}[0, 1]$. This can be proved by induction, since $\tilde{z}u_B^{\frac{1}{\alpha+B-1}} \sim \text{Gamma}(\alpha + B - 1, 1)$, $\tilde{z}u_B^{\frac{1}{\alpha+B-1}}u_{B-1}^{\frac{1}{\alpha+B-2}} \sim \text{Gamma}(\alpha + B - 2, 1)$, etc. Hence, we can apply the rejection sampling framework for $\tilde{z} \sim \text{Gamma}(\alpha + B, 1)$ instead of the original z. We study the effect of shape augmentation on the variance in Section 5.2.

5.2 Dirichlet Distribution

The $\text{Dirichlet}(\alpha_{1:K})$ distribution, with concentration parameters $\alpha_{1:K}$, is a K-dimensional multivariate distribution with $K - 1$ degrees of free-

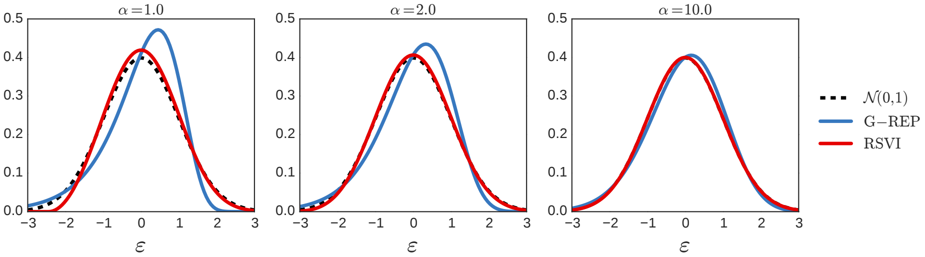

Figure 3: In the distribution on the transformed space ε for a gamma distribution we can see that the rejection sampling-inspired transformation converges faster to a standard normal. Therefore it is less dependent on the parameter α, which implies a smaller correction term. We compare the transformation of RSVI (this paper) with the standardization procedure suggested in G-REP [Ruiz et al., 2016], for shape parameters $\alpha = \{1, 2, 10\}$.

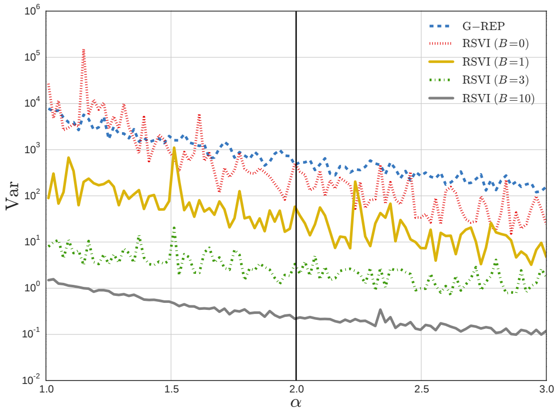

Figure 4: RSVI (this paper) achieves lower variance compared to G-REP [Ruiz et al., 2016]. The estimated variance is for the first component of Dirichlet approximation to a multinomial likelihood with uniform Dirichlet prior. Optimal concentration is $\alpha = 2$, and B denotes shape augmentation.

dom. To simulate random variables we use the fact that if $\tilde{z}_k \sim \text{Gamma}(\alpha_k, 1)$ i.i.d., then $z_{1:K} = \left(\sum_\ell \tilde{z}_\ell\right)^{-1} (\tilde{z}_1, \dots, \tilde{z}_K)^\top \sim \text{Dirichlet}(\alpha_{1:K})$.

Thus, we make a change of variables to reduce the problem to that of simulating independent gamma distributed random variables,

$$\mathbb{E}_{q(z_{1:K}\,;\alpha_{1:K})}[f(z_{1:K})] =$$

$$= \int f\left(\frac{\tilde{z}_{1:K}}{\sum_{\ell=1}^K \tilde{z}_\ell}\right) \prod_{k=1}^K \text{Gamma}(\tilde{z}_k\,; \alpha_k, 1) \mathrm{d}\tilde{z}_{1:K}.$$

We apply the transformation in Section 5.1 for the gamma-distributed variables, $\tilde{z}_k = h_{\text{Gamma}}(\varepsilon_k, \alpha_k)$, where the variables ε_k are generated by independent gamma rejection samplers. To showcase this, we study a simple conjugate model where the exact gradient and posterior are available: a multinomial likelihood with Dirichlet prior and Dirichlet variational distribution. In Figure 4 we show the resulting variance of the first component of the gradient, based on simulated data from a Dirichlet distribution with $K = 100$ components, uniform prior, and $N = 100$ trials. We compare the variance of RSVI (for various shape augmentation settings) with the G-REP approach [Ruiz et al., 2016]. RSVI performs better even without the augmentation trick, and significantly better with it.

6 Experiments

In Section 5 we compared rejection sampling variational inference (RSVI) with generalized reparameterization (G-REP) and found a substantial variance reduction on synthetic examples. Here we evaluate RSVI on a more challenging model, the sparse gamma deep exponential family (DEF) [Ranganath et al., 2015]. On two real datasets, we compare RSVI with state-of-the-art methods: automatic differentiation variational inference (ADVI) [Kucukelbir et al., 2015, 2016], black-box variational inference (BBVI) [Ranganath et al., 2014], and G-REP [Ruiz et al., 2016].

Data. The datasets we consider are the Olivetti faces[2] and Neural Information Processing Systems (NIPS) 2011 conference papers. The Olivetti faces dataset consists of 64×64 gray-scale images of human faces in 8 bits, i.e., the data is discrete and in the set $\{0, \dots, 255\}$. In the NIPS dataset we have documents in a bag-of-words format with an effective vocabulary of 5715 words.

Model. The sparse gamma DEF [Ranganath et al., 2015] is a multi-layered probabilistic model that mimics the architecture of deep neural networks. It models the data using a set of local latent variables $z_{n,k}^\ell$ where n indexes observations, k components, and ℓ layers. These local variables are connected between layers through global weights $w_{k,k'}^\ell$. The observations are $x_{n,d}$, where d denotes dimension. The joint proba-

[2] `http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html`

	RSVI $B = 1$	RSVI $B = 4$	G-REP
Min	6.0e−4	1.2e−3	2.7e−3
Median	**9.0e7**	**2.9e7**	1.6e12
Max	1.2e17	**3.4e14**	1.5e17

	RSVI $B = 1$	RSVI $B = 4$	G-REP
Min	1.8e−3	1.5e−3	2.6e−3
Median	1.2e4	**4.5e3**	1.5e7
Max	1.4e12	**1.6e11**	3.5e12

Table 1: The RSVI gradient (this paper) exhibits lower variance than G-REP [Ruiz et al., 2016]. We show estimated variance, based on 10 samples, of G-REP and RSVI (for $B = 1, 4$ shape augmentation steps), for parameters at the initialization point (*left*) and at iteration 2600 in RSVI (*right*), estimated for the NIPS data.

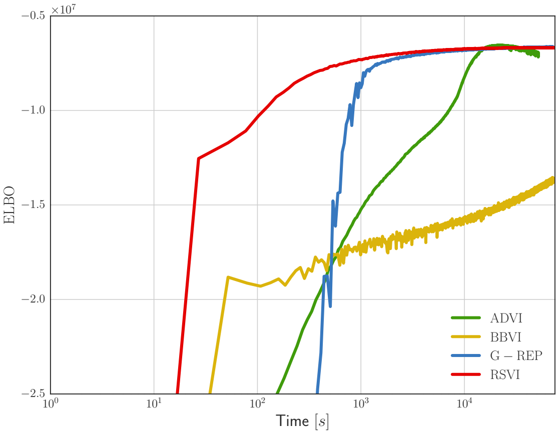

Figure 5: RSVI (this paper) presents a significantly faster initial improvement of the evidence lower bound (ELBO) as a function of wall-clock time. The model is a sparse gamma DEF, applied to the Olivetti faces dataset, and we compare with ADVI [Kucukelbir et al., 2016], BBVI [Ranganath et al., 2014], and G-REP [Ruiz et al., 2016].

bilistic model is defined as

$$\begin{aligned} z_{n,k}^{\ell} &\sim \text{Gamma}\left(\alpha_z, \frac{\alpha_z}{\sum_{k'} w_{k,k'}^{\ell} z_{n,k'}^{\ell+1}}\right), \\ x_{n,d} &\sim \text{Poisson}\left(\sum_k w_{k,d}^{0} z_{n,k}^{1}\right). \end{aligned} \tag{11}$$

We set $\alpha_z = 0.1$ in the experiments. All priors on the weights are set to $\text{Gamma}(0.1, 0.3)$, and the top-layer local variables priors are set to $\text{Gamma}(0.1, 0.1)$. We use 3 layers, with 100, 40, and 15 components in each. This is the same model that was studied by Ruiz et al. [2016], where G-REP was shown to outperform both BBVI (with control variates and Rao-Blackwellization), as well as ADVI. In the experiments we follow their approach and parameterize the variational approximating gamma distribution using the shape and mean. To avoid constrained optimization we use the transform $\theta = \log(1+\exp(\vartheta))$ for non-negative variational parameters θ, and optimize ϑ in the unconstrained space.

Results. For the Olivetti faces we explore $\eta \in \{0.75, 1, 2, 5\}$ and show the resulting ELBO of the best one in Figure 5. We can see that RSVI has a significantly faster initial improvement than any of the other methods.[3] The wall-clock time for RSVI is based on a Python implementation (average 1.5s per iteration) using the automatic differentiation package autograd [Maclaurin et al., 2015]. We found that RSVI is approximately two times faster than G-REP for comparable implementations. One reason for this is that the transformations based on rejection sampling are cheaper to evaluate. Indeed, the research literature on rejection sampling is heavily focused on finding cheap and efficient transformations.

For the NIPS dataset, we now compare the variance of the gradients between the two estimators, RSVI and G-REP, for different shape augmentation steps B. In Table 1 we show the minimum, median, and maximum values of the variance across all dimensions. We can see that RSVI again clearly outperforms G-REP in terms of variance. Moreover, increasing the number of augmentation steps B provides even further improvements.

7 Conclusions

We introduced rejection sampling variational inference (RSVI), a method for deriving reparameterization gradients when simulation from the variational distribution is done using a acceptance-rejection sampler. In practice, RSVI leads to lower-variance gradients than other state-of-the-art methods. Further, it enables reparameterization gradients for a large class of variational distributions, taking advantage of the efficient transformations developed in the rejection sampling literature.

This work opens the door to other strategies that "remove the lid" from existing black-box samplers in the service of variational inference. As future work, we can consider more complicated simulation algorithms with accept-reject-like steps, such as adaptive rejection sampling, importance sampling, sequential Monte Carlo, or Markov chain Monte Carlo.

[3] The results of G-REP, ADVI and BBVI where reproduced with permission from Ruiz et al. [2016].

Acknowledgements

Christian A. Naesseth is supported by CADICS, a Linnaeus Center, funded by the Swedish Research Council (VR). Francisco J. R. Ruiz is supported by the EU H2020 programme (Marie Skłodowska-Curie grant agreement 706760). Scott W. Linderman is supported by the Simons Foundation SCGB-418011. This work is supported by NSF IIS-1247664, ONR N00014-11-1-0651, DARPA PPAML FA8750-14-2-0009, DARPA SIMPLEX N66001-15-C-4032, Adobe, and the Alfred P. Sloan Foundation. The authors would like to thank Alp Kucukelbir and Dustin Tran for helpful comments and discussion.

References

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *arXiv:1601.00670*, 2016.

G. Bonnet. Transformations des signaux aléatoires a travers les systemes non linéaires sans mémoire. *Annals of Telecommunications*, 19(9):203–220, 1964.

G. Casella and C. P. Robert. Rao-Blackwellisation of sampling schemes. *Biometrika*, 83(1):81–94, 1996.

L. Devroye. *Non-Uniform Random Variate Generation.* Springer-Verlag, 1986.

J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, jul 2011.

K. Fan, Z. Wang, J. Beck, J. Kwok, and K. A. Heller. Fast second order stochastic backpropagation for variational inference. In *Advances in Neural Information Processing Systems*, 2015.

P. W. Glynn. Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33(10):75–84, oct 1990.

G. E. Hinton and D. van Camp. Keeping the neural networks simple by minimizing the description length of the weights. In *Proceedings of the Sixth Annual Conference on Computational Learning Theory*, pages 5–13, New York, NY, USA, 1993. ACM.

M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347, May 2013.

E. Jang, S. Gu, and B. Poole. Categorical reparameterization using gumbel-softmax. In *International Conference on Learning Representations*, 2017. (accepted for publication).

M. I. Jordan, Z. Ghahramani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2): 183–233, Nov. 1999.

D. P. Kingma and M. Welling. Auto-encoding variational Bayes. In *International Conference on Learning Representations*, 2014.

D. A. Knowles. Stochastic gradient variational Bayes for Gamma approximating distributions. *arXiv:1509.01631v1*, 2015.

A. Kucukelbir, R. Ranganath, A. Gelman, and D. M. Blei. Automatic variational inference in Stan. In *Advances in Neural Information Processing Systems*, 2015.

A. Kucukelbir, D. Tran, R. Ranganath, A. Gelman, and D. M. Blei. Automatic differentiation variational inference. *arXiv:1603.00788*, 2016.

L. Maaløe, C. K. Sønderby, S. K. Sønderby, and O. Winther. Auxiliary deep generative models. In *International Conference on Machine Learning*, 2016.

D. Maclaurin, D. Duvenaud, M. Johnson, and R. P. Adams. Autograd: Reverse-mode differentiation of native Python, 2015. URL `http://github.com/HIPS/autograd`.

C. J. Maddison, A. Mnih, and Y. W. Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *International Conference on Learning Representations*, 2017. (accepted for publication).

G. Marsaglia and W. W. Tsang. A simple method for generating gamma variables. *ACM Transactions on Mathematical Software*, 26(3):363–372, Sept. 2000.

A. Mnih and K. Gregor. Neural variational inference and learning in belief networks. In *International Conference on Machine Learning*, 2014.

J. W. Paisley, D. M. Blei, and M. I. Jordan. Variational Bayesian inference with stochastic search. In *International Conference on Machine Learning*, 2012.

O. Papaspiliopoulos, G. O. Roberts, and M. Sköld. Non-centered parameterisations for hierarchical models and data augmentation. In *Bayesian Statistics 7: Proceedings of the Seventh Valencia International Meeting*, page 307. Oxford University Press, USA, 2003.

R. Price. A useful theorem for nonlinear devices having Gaussian inputs. *IRE Transactions on Information Theory*, 4(2):69–72, 1958.

R. Ranganath, S. Gerrish, and D. M. Blei. Black box variational inference. In *Artificial Intelligence and Statistics*, 2014.

R. Ranganath, L. Tang, L. Charlin, and D. M. Blei. Deep exponential families. In *Artificial Intelligence and Statistics*, 2015.

R. Ranganath, D. Tran, and D. M. Blei. Hierarchical variational models. In *International Conference on Machine Learning*, 2016.

D. J. Rezende, S. Mohamed, and D. Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, 2014.

C. Robert and G. Casella. *Monte Carlo statistical methods*. Springer Science & Business Media, 2004.

F. J. R. Ruiz, M. K. Titsias, and D. M. Blei. The generalized reparameterization gradient. In *Advances in Neural Information Processing Systems*, 2016.

T. Salimans and D. A. Knowles. Fixed-form variational posterior approximation through stochastic linear regression. *Bayesian Analysis*, 8(4):837–882, 2013.

T. Salimans, D. P. Kingma, and M. Welling. Markov chain Monte Carlo and variational inference: Bridging the gap. In *International Conference on Machine Learning*, 2015.

J. Schulman, N. Heess, T. Weber, and P. Abbeel. Gradient estimation using stochastic computation graphs. In *Advances in Neural Information Processing Systems*, 2015.

A. Stuart. Gamma-distributed products of independent random variables. *Biometrika*, 49:64–65, 1962.

T. Tieleman and G. Hinton. Lecture 6.5-RMSPROP: Divide the gradient by a running average of its recent magnitude. Coursera: Neural Networks for Machine Learning, 4, 2012.

M. K. Titsias and M. Lázaro-Gredilla. Doubly stochastic variational Bayes for non-conjugate inference. In *International Conference on Machine Learning*, 2014.

D. Tran, R. Ranganath, and D. M. Blei. The variational Gaussian process. In *International Conference on Learning Representations*, 2016.

J. von Neumann. Various Techniques Used in Connection with Random Digits. *Journal of Research of the National Bureau of Standards*, 12:36–38, 1951.

S. Waterhouse, D. Mackay, and T. Robinson. Bayesian methods for mixtures of experts. In *Advances in Neural Information Processing Systems*, pages 351–357. MIT Press, 1996.

R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3–4):229–256, 1992.

A Supplementary Material

A.1 Distribution of ε

Here we formalize the claim in the main manuscript regarding the distribution of the accepted variable ε in the rejection sampler. Recall that $z = h(\varepsilon, \theta)$, $\varepsilon \sim s(\varepsilon)$ is equivalent to $z \sim r(z\,;\theta)$, and that $q(z\,;\theta) \le M_\theta r(z\,;\theta)$. For simplicity we consider the univariate continuous case in the exposition below, but the result also holds for the discrete and multivariate settings. The cumulative distribution function for the accepted ε is given by

$$
\begin{aligned}
&\mathbb{P}(E \le \varepsilon) = \sum_{i=1}^{\infty} \mathbb{P}(E \le \varepsilon, E = E_i) \\
&= \sum_{i=1}^{\infty} \Bigg[\mathbb{P}\left(E_i \le \varepsilon, U_i < \frac{q(h(E_i,\theta)\,;\theta)}{M_\theta r(h(E_i,\theta)\,;\theta)} \right) \\
&\quad \prod_{j=1}^{i-1} \mathbb{P}\left(U_j \ge \frac{q(h(E_j,\theta)\,;\theta)}{M_\theta r(h(E_j,\theta)\,;\theta)} \right) \Bigg] \\
&= \sum_{i=1}^{\infty} \int_{-\infty}^{\varepsilon} s(e) \frac{q(h(e,\theta)\,;\theta)}{M_\theta r(h(e,\theta)\,;\theta)} \mathrm{d}e \prod_{j=1}^{i-1} \left(1 - \frac{1}{M_\theta} \right) \\
&= \int_{-\infty}^{\varepsilon} s(e) \frac{q(h(e,\theta)\,;\theta)}{r(h(e,\theta)\,;\theta)} \mathrm{d}e \cdot \frac{1}{M_\theta} \cdot \sum_{i=1}^{\infty} \left(1 - \frac{1}{M_\theta} \right)^{i-1} \\
&= \int_{-\infty}^{\varepsilon} s(e) \frac{q(h(e,\theta)\,;\theta)}{r(h(e,\theta)\,;\theta)} \mathrm{d}e.
\end{aligned}
$$

Here, we have applied that $z = h(\varepsilon, \theta)$, $\varepsilon \sim s(\varepsilon)$ is a reparameterization of $z \sim r(z\,;\theta)$, and thus

$$
\begin{aligned}
&\mathbb{P}\left(U_j \ge \frac{q(h(E_j,\theta)\,;\theta)}{M_\theta r(h(E_j,\theta)\,;\theta)} \right) \\
&= \int_{-\infty}^{\infty} s(e) \left(1 - \frac{q(h(e,\theta)\,;\theta)}{M_\theta r(h(e,\theta)\,;\theta)} \right) \mathrm{d}e \\
&= 1 - \frac{1}{M_\theta} \mathbb{E}_{s(e)} \left[\frac{q(h(e,\theta)\,;\theta)}{r(h(e,\theta)\,;\theta)} \right] \\
&= 1 - \frac{1}{M_\theta} \mathbb{E}_{r(z\,;\theta)} \left[\frac{q(z\,;\theta)}{r(z\,;\theta)} \right] = 1 - \frac{1}{M_\theta}.
\end{aligned}
$$

The density is obtained by taking the derivative of the cumulative distribution function with respect to ε,

$$
\frac{\mathrm{d}}{\mathrm{d}\varepsilon} \mathbb{P}(E \le \varepsilon) = s(\varepsilon) \frac{q(h(\varepsilon,\theta)\,;\theta)}{r(h(\varepsilon,\theta)\,;\theta)},
$$

which is the expression from the main manuscript.

The motivation from the main manuscript is basically a standard "area-under-the-curve" or geometric argument for rejection sampling [Robert and Casella, 2004], but for ε instead of z.

A.2 Derivation of the Gradient

We provide below details for the derivation of the gradient. We assume that h is differentiable (almost everywhere) with respect to θ, and that $f(h(\varepsilon,\theta)) \frac{q(h(\varepsilon,\theta)\,;\theta)}{r(h(\varepsilon,\theta)\,;\theta)}$ is continuous in θ for all ε. Then, we have

$$
\begin{aligned}
&\nabla_\theta \mathbb{E}_{q(z\,;\theta)}[f(z)] = \nabla_\theta \mathbb{E}_{\pi(\varepsilon\,;\theta)}[f(h(\varepsilon,\theta))] \\
&= \int s(\varepsilon) \nabla_\theta \left(f\left(h(\varepsilon,\theta)\right) \frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)} \right) \mathrm{d}\varepsilon \\
&= \int s(\varepsilon) \frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)} \nabla_\theta f\left(h(\varepsilon,\theta)\right) \mathrm{d}\varepsilon \\
&+ \int s(\varepsilon) f\left(h(\varepsilon,\theta)\right) \nabla_\theta \left(\frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)} \right) \mathrm{d}\varepsilon \\
&= \underbrace{\mathbb{E}_{\pi(\varepsilon\,;\theta)}[\nabla_\theta f(h(\varepsilon,\theta))]}_{=:g_{\mathrm{rep}}} + \\
&\quad + \underbrace{\mathbb{E}_{\pi(\varepsilon\,;\theta)}\left[f(h(\varepsilon,\theta)) \nabla_\theta \log \frac{q(h(\varepsilon,\theta)\,;\theta)}{r(h(\varepsilon,\theta)\,;\theta)} \right]}_{=:g_{\mathrm{cor}}},
\end{aligned}
$$

where in the last step we have identified $\pi(\varepsilon\,;\theta)$ and made use of the log-derivative trick

$$
\nabla_\theta \frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)} = \frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)} \nabla_\theta \log \frac{q\left(h(\varepsilon,\theta)\,;\theta\right)}{r\left(h(\varepsilon,\theta)\,;\theta\right)}.
$$

Gradient of Log-Ratio in $g_{\mathbf{cor}}$ For invertible reparameterizations we can simplify the evaluation of the gradient of the log-ratio in g_{cor} as follows using standard results on transformation of a random variable

$$
\begin{aligned}
&\nabla_\theta \log \frac{q(h(\varepsilon,\theta)\,;\theta)}{r(h(\varepsilon,\theta)\,;\theta)} = \nabla_\theta \log q(h(\varepsilon,\theta)\,;\theta) + \\
&+ \nabla_\theta \log \left| \frac{dh}{d\varepsilon}(\varepsilon,\theta) \right| - \nabla_\theta \log \underbrace{s(h^{-1}(h(\varepsilon,\theta),\theta))}_{=\ s(\varepsilon)} \\
&= \nabla_\theta \log q(h(\varepsilon,\theta)\,;\theta) + \nabla_\theta \log \left| \frac{dh}{d\varepsilon}(\varepsilon,\theta) \right|.
\end{aligned}
$$

A.3 Examples of Reparameterizable Rejection Samplers

We show in Table 2 some examples of reparameterizable rejection samplers for three distributions, namely, the gamma, the truncated normal, and the von Misses distributions (for more examples, see Devroye [1986]). We show the distribution $q(z\,;\theta)$, the transformation $h(\varepsilon,\theta)$, and the proposal $s(\varepsilon)$ used in the rejection sampler.

We show in Table 3 six examples of distributions that can be reparameterized in terms of auxiliary gamma-distributed random variables. We show the distribution $q(z\,;\theta)$, the distribution of the auxiliary gamma random variables $p(\tilde{z}\,;\theta)$, and the mapping $z = g(\tilde{z},\theta)$.

$q(z\,;\theta)$	$h(\varepsilon,\theta)$	$s(\varepsilon)$
$\text{Gamma}(\alpha,1)$	$\left(\alpha-\frac{1}{3}\right)\left(1+\frac{\varepsilon}{\sqrt{9\alpha-3}}\right)^3$	$\varepsilon\sim\mathcal{N}(0,1)$
Truncated $\mathcal{N}(0,1,a,\infty)$	$\sqrt{a^2-2\log\varepsilon}$	$\varepsilon\sim\mathcal{U}[0,1]$
$\text{vonMises}(\kappa)$	$\frac{\text{sign}(\varepsilon)}{\cos\left(\frac{1+c\cos(\pi\varepsilon)}{c+\cos(\pi\varepsilon)}\right)}$, $c=\frac{1+\rho^2}{2\rho}$, $\rho=\frac{r-\sqrt{2r}}{2\kappa}$, $r=1+\sqrt{1+4\kappa^2}$	$\varepsilon\sim\mathcal{U}[-1,1]$

Table 2: Examples of reparameterizable rejection samplers; many more can be found in Devroye [1986]. The first column is the distribution, the second column is the transformation $h(\varepsilon,\theta)$, and the last column is the proposal $s(\varepsilon)$.

$q(z\,;\theta)$	$g(\tilde{z},\theta)$	$p(\tilde{z}\,;\theta)$
$\text{Beta}(\alpha,\beta)$	$\dfrac{\tilde{z}_1}{\tilde{z}_1+\tilde{z}_2}$	$\tilde{z}_1\sim\text{Gamma}(\alpha,1)$, $\tilde{z}_2\sim\text{Gamma}(\beta,1)$
$\text{Dirichlet}(\alpha_{1:K})$	$\dfrac{1}{\sum_\ell\tilde{z}_\ell}\left(\tilde{z}_1,\ldots,\tilde{z}_K\right)^\top$	$\tilde{z}_k\sim\text{Gamma}(\alpha_k,1),\ k=1,\ldots,K$
$\text{St}(\nu)$	$\sqrt{\dfrac{\nu}{2\tilde{z}_1}}\tilde{z}_2$	$\tilde{z}_1\sim\text{Gamma}(\nu/2,1)$, $\tilde{z}_2\sim\mathcal{N}(0,1)$
$\chi^2(k)$	$2\tilde{z}$	$\tilde{z}\sim\text{Gamma}(k/2,1)$
$\text{F}(d_1,d_2)$	$\dfrac{d_2\tilde{z}_1}{d_1\tilde{z}_2}$	$\tilde{z}_1\sim\text{Gamma}(d_1/2,1)$, $\tilde{z}_2\sim\text{Gamma}(d_2/2,1)$
$\text{Nakagami}(m,\Omega)$	$\sqrt{\dfrac{\Omega\tilde{z}}{m}}$	$\tilde{z}\sim\text{Gamma}(m,1)$

Table 3: Examples of random variables as functions of auxiliary random variables with reparameterizable distributions. The first column is the distribution, the second column is a function $g(\tilde{z},\theta)$ mapping from the auxiliary variables to the desired variable, and the last column is the distribution of the auxiliary variables $\tilde{z}$.

A.4 Reparameterizing the Gamma Distribution

We provide details on reparameterization of the gamma distribution. In the following we consider rate $\beta=1$. Note that this is not a restriction, we can always reparameterize the rate. The density of the gamma random variable is given by

$$q(z\,;\alpha)=\frac{z^{\alpha-1}e^{-z}}{\Gamma(\alpha)},$$

where $\Gamma(\alpha)$ is the gamma function. We make use of the reparameterization defined by

$$z=h(\varepsilon,\alpha)=\left(\alpha-\frac{1}{3}\right)\left(1+\frac{\varepsilon}{\sqrt{9\alpha-3}}\right)^3,$$
$$\varepsilon\sim\mathcal{N}(0,1).$$

Because h is invertible we can make use of the simplified gradient of the log-ratio derived in Section A.2 above. The gradients of $\log q$ and $-\log r$ are given by

$$\begin{aligned}&\nabla_\alpha\log q(h(\varepsilon,\alpha)\,;\alpha)\\&\quad=\log(h(\varepsilon,\alpha))+(\alpha-1)\frac{\frac{dh(\varepsilon,\alpha)}{d\alpha}}{h(\varepsilon,\alpha)}-\frac{dh(\varepsilon,\alpha)}{d\alpha}-\psi(\alpha),\\&\nabla_\alpha-\log r(h(\varepsilon,\alpha)\,;\alpha)=\nabla_\alpha\log\left|\frac{dh}{d\varepsilon}(\varepsilon,\alpha)\right|\\&\quad=\frac{1}{2\left(\alpha-\frac{1}{3}\right)}-\frac{9\varepsilon}{\left(1+\frac{\varepsilon}{\sqrt{9\alpha-3}}\right)(9\alpha-3)^{\frac{3}{2}}},\end{aligned}$$

where $\psi(\alpha)$ is the digamma function and

$$\begin{aligned}&\frac{dh(\varepsilon,\alpha)}{d\alpha}\\&\quad=\left(1+\frac{\varepsilon}{\sqrt{9\alpha-3}}\right)^3-\frac{27\varepsilon}{2(9\alpha-3)^{\frac{3}{2}}}\left(1+\frac{\varepsilon}{\sqrt{9\alpha-3}}\right)^2.\end{aligned}$$