Overdispersed Black-Box Variational Inference

Francisco J. R. Ruiz
Data Science Institute
Dept. of Computer Science
Columbia University

Michalis K. Titsias
Dept. of Informatics
Athens University of
Economics and Business

David M. Blei
Data Science Institute
Dept. of Computer Science and Statistics
Columbia University

Abstract

We introduce overdispersed black-box variational inference, a method to reduce the variance of the Monte Carlo estimator of the gradient in black-box variational inference. Instead of taking samples from the variational distribution, we use importance sampling to take samples from an overdispersed distribution in the same exponential family as the variational approximation. Our approach is general since it can be readily applied to any exponential family distribution, which is the typical choice for the variational approximation. We run experiments on two non-conjugate probabilistic models to show that our method effectively reduces the variance, and the overhead introduced by the computation of the proposal parameters and the importance weights is negligible. We find that our overdispersed importance sampling scheme provides lower variance than black-box variational inference, even when the latter uses twice the number of samples. This results in faster convergence of the black-box inference procedure.

1 INTRODUCTION

Generative probabilistic modeling is an effective approach for understanding real-world data in many areas of science (Bishop, 2006; Murphy, 2012). A probabilistic model describes a data-generating process through a joint distribution of observed data and latent (unobserved) variables. With a model in place, the investigator uses an inference algorithm to calculate or approximate the posterior, i.e., the conditional distribution of the latent variables given the available observations. It is through the posterior that the investigator explores the latent structure in the data and forms a predictive distribution of future data. Approximating the posterior is the central algorithmic problem for probabilistic modeling.

One of the most widely used methods to approximate the posterior distribution is variational inference (Wainwright and Jordan, 2008; Jordan et al., 1999). Variational inference aims to approximate the posterior with a simpler distribution, fitting that distribution to be close to the exact posterior, where closeness is measured in terms of Kullback-Leibler (KL) divergence. In minimizing the KL, variational inference converts the problem of approximating the posterior into an optimization problem.

Traditional variational inference uses coordinate ascent to optimize its objective. This works well for models in which each conditional distribution is easy to compute (Ghahramani and Beal, 2001), but is difficult to use in more complex models where the variational objective involves intractable expectations. Recent innovations in variational inference have addressed this with stochastic optimization, forming noisy gradients with Monte Carlo approximation. This strategy expands the scope of variational inference beyond traditional models, e.g., to non-conjugate probabilistic models (Carbonetto et al., 2009; Paisley et al., 2012; Salimans and Knowles, 2013; Ranganath et al., 2014; Titsias and Lázaro-Gredilla, 2014), deep neural networks (Neal, 1992; Hinton et al., 1995; Mnih and Gregor, 2014; Kingma and Welling, 2014; Ranganath et al., 2015), and probabilistic programming (Wingate and Weber, 2013; Kucukelbir et al., 2015). Some of these techniques find their roots in classical policy search algorithms for reinforcement learning (Williams, 1992; van de Meent et al., 2016).

These approaches must address a core problem with Monte Carlo estimates of the gradient, which is that they suffer from high variance. The estimated gradient can significantly differ from the truth and this leads to slow convergence of the optimization. There are several strategies to reduce the variance of the gradients, including Rao-Blackwellization (Casella and Robert, 1996; Ranganath et al., 2014), control variates (Ross, 2002; Paisley et al., 2012; Ranganath et al., 2014; Gu et al., 2016), reparameterization (Price, 1958; Bonnet, 1964; Salimans and Knowles, 2013; Kingma and Welling, 2014; Rezende et al., 2014; Kucukelbir et al., 2015), and local expectations (Titsias and

Lázaro-Gredilla, 2015).

In this paper we develop overdispersed black-box variational inference (O-BBVI), a new method for reducing the variance of Monte Carlo gradients in variational inference. The main idea is to use importance sampling to estimate the gradient, in order to construct a good proposal distribution that is matched to the variational problem. We show that O-BBVI applies more generally than methods such as reparameterization and local expectations, and it further improves the profile of gradients that use Rao-Blackwellization and control variates.

We demonstrate O-BBVI on two complex models: a non-conjugate time series model (Ranganath et al., 2014) and Poisson-based deep exponential families (DEFs) (Ranganath et al., 2015). Our study shows that O-BBVI reduces the variance of the original black-box variational inference (BBVI) estimates (Ranganath et al., 2014), even when using only half the number of Monte Carlo samples. This provides significant savings in run-time complexity.

Technical summary. Consider a probabilistic model $p(\mathbf{x}, \mathbf{z})$, where $\mathbf{z}$ are the latent variables and $\mathbf{x}$ are the observations. Variational inference sets up a parameterized distribution of the latent variables $q(\mathbf{z}; \boldsymbol{\lambda})$ and finds the parameter $\boldsymbol{\lambda}^\star$ that minimizes the KL divergence between $q(\mathbf{z}; \boldsymbol{\lambda})$ and the posterior $p(\mathbf{z} \mid \mathbf{x})$. We then use $q(\mathbf{z}; \boldsymbol{\lambda}^\star)$ as a proxy for the posterior.

We build on BBVI, which solves this problem with a stochastic optimization procedure that uses Monte Carlo estimates of the gradient (Ranganath et al., 2014). Let $\mathcal{L}(\boldsymbol{\lambda})$ be the variational objective, which is the (negative) KL divergence up to an additive constant. BBVI uses samples from $q(\mathbf{z}; \boldsymbol{\lambda})$ to approximate its gradient,

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L} = \mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})} \left[f(\mathbf{z}) \right], \tag{1}$$

where

$$f(\mathbf{z}) = \nabla_{\boldsymbol{\lambda}} \log q(\mathbf{z}; \boldsymbol{\lambda}) \left(\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}; \boldsymbol{\lambda}) \right). \tag{2}$$

The resulting Monte Carlo estimator, based on sampling from $q(\mathbf{z}; \boldsymbol{\lambda})$, only requires evaluating the log-joint distribution $\log p(\mathbf{z}, \mathbf{x})$, the log-variational distribution $\log q(\mathbf{z}; \boldsymbol{\lambda})$, and the score function $\nabla_{\boldsymbol{\lambda}} \log q(\mathbf{z}; \boldsymbol{\lambda})$. Calculations about $q(\mathbf{z}; \boldsymbol{\lambda})$ can be derived once and stored in a library and, as a consequence, BBVI can be easily applied to a large class of models. However, as we mentioned above, Monte Carlo estimates of this gradient usually have high variance. Ranganath et al. (2014) correct for this with Rao-Blackwellization and control variates.

We expand on this idea by approximating the gradient with importance sampling. We introduce a proposal distribution $r(\mathbf{z}; \boldsymbol{\lambda}, \tau)$, which depends on both the variational parameters and an additional parameter. (We discuss the additional parameter below.) We then write the gradient as

$$\nabla_{\boldsymbol{\lambda}} \mathcal{L} = \mathbb{E}_{r(\mathbf{z};\boldsymbol{\lambda},\tau)} \left[f(\mathbf{z}) \frac{q(\mathbf{z}; \boldsymbol{\lambda})}{r(\mathbf{z}; \boldsymbol{\lambda}, \tau)} \right], \tag{3}$$

and form noisy estimates with samples from the proposal.

The key idea behind our method is that the optimal proposal distribution (in terms of minimizing the variance of the resulting estimator) is *not* the original distribution $q(\mathbf{z}; \boldsymbol{\lambda})$ (Owen, 2013, Chapter 9). Rather, the optimal proposal is a skewed version of that distribution with heavier tails. Unfortunately, this distribution is not available to us—it involves an intractable normalization constant. But we use this insight to set a proposal with heavier tails than the variational distribution, thus making it closer to the optimal proposal. Note this is an unconventional use of importance sampling, which is usually employed to approximate expectations. Instead, we use importance sampling to improve the characteristics of a Monte Carlo estimator by sampling from a different distribution.

In detail, we first assume that the variational distribution is in the exponential family. (This is not an assumption about the model; most applications of variational inference use exponential family variational distributions.) We then set the proposal distribution to be in the corresponding overdispersed exponential family (Jørgensen, 1987), where τ is the dispersion parameter. We show that the corresponding estimator has lower variance than the BBVI estimator, we put forward a method to adapt the dispersion parameter during optimization, and we demonstrate that this method is more efficient than BBVI. We call our approach *overdispersed black-box variational inference* (O-BBVI).

Organization. The rest of the paper is organized as follows. We review BBVI in Section 2. We develop O-BBVI in Section 3, describing both the basic algorithm and its extensions to adaptive proposals and high-dimensional settings. Section 4 reports on our empirical study of two non-conjugate models. We conclude the paper in Section 5.

2 BLACK-BOX VARIATIONAL INFERENCE

Consider a probabilistic model $p(\mathbf{x}, \mathbf{z})$ and a variational family $q(\mathbf{z}; \boldsymbol{\lambda})$ which is in the exponential family, i.e.,

$$q(\mathbf{z}; \boldsymbol{\lambda}) = g(\mathbf{z}) \exp\left\{ \boldsymbol{\lambda}^\top t(\mathbf{z}) - A(\boldsymbol{\lambda}) \right\}, \tag{4}$$

where $g(\mathbf{z})$ is the base measure, $\boldsymbol{\lambda}$ are the natural parameters, $t(\mathbf{z})$ are the sufficient statistics, and $A(\boldsymbol{\lambda})$ is the log-normalizer. We are interested in a variational approximation to the intractable posterior $p(\mathbf{z} \mid \mathbf{x})$, i.e., we aim to minimize the KL divergence $D_{\mathrm{KL}}\left(q(\mathbf{z}; \boldsymbol{\lambda}) \parallel p(\mathbf{z} \mid \mathbf{x})\right)$ with respect to $\boldsymbol{\lambda}$ (Jordan et al., 1999). This is equivalent to maximizing the evidence lower bound (ELBO),

$$\mathcal{L}(\boldsymbol{\lambda}) = \mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})} \left[\log p(\mathbf{x}, \mathbf{z}) - \log q(\mathbf{z}; \boldsymbol{\lambda}) \right], \tag{5}$$

which is a lower bound on the log of the marginal probability of the observations, $\log p(\mathbf{x})$.

With a tractable variational family (e.g., the mean-field family) and a conditionally conjugate model,[1] the expectations in Eq. 5 can be computed in closed form and we can use coordinate-ascent variational inference (Ghahramani and Beal, 2001). However, many models of interest are not conditionally conjugate. For these models, we need alternative methods to optimize the ELBO. One approach is BBVI, which uses Monte Carlo estimates of the gradient and requires few model-specific calculations (Ranganath et al., 2014). Thus, BBVI is a variational inference algorithm that can be applied to a large class of models.

BBVI relies on the "log-derivative trick," also called REINFORCE or score function method (Williams, 1992; Kleijnen and Rubinstein, 1996; Glynn, 1990), to obtain Monte Carlo estimates of the gradient. In detail, we recover the Monte Carlo estimate driven by Eqs. 1 and 2 by taking the gradient of (5) with respect to the variational parameters $\boldsymbol{\lambda}$, and then applying the following two identities:

$$\nabla_{\boldsymbol{\lambda}} q(\mathbf{z};\boldsymbol{\lambda}) = q(\mathbf{z};\boldsymbol{\lambda}) \nabla_{\boldsymbol{\lambda}} \log q(\mathbf{z};\boldsymbol{\lambda}), \tag{6}$$

$$\mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})} \left[\nabla_{\boldsymbol{\lambda}} \log q(\mathbf{z};\boldsymbol{\lambda})\right] = 0. \tag{7}$$

Eq. 1 enables noisy gradients of the ELBO by taking samples from $q(\mathbf{z};\boldsymbol{\lambda})$. However, the resulting estimator may have high variance. This is especially the case when the variational distribution $q(\mathbf{z};\boldsymbol{\lambda})$ is a poor fit to the posterior $p(\mathbf{z} \mid \mathbf{x})$, which is typical in early iterations of optimization.

In order to reduce the variance of the estimator, BBVI uses two strategies: control variates and Rao-Blackwellization. Because we will also use these ideas in our algorithm, we briefly discuss them here.

Control variates. A control variate is a random variable that is included in the estimator, preserving its expectation but reducing its variance (Ross, 2002). Although there are many possible choices for control variates, Ranganath et al. (2014) advocate for the weighted score function because it is not model-dependent. Denote the score function by $h(\mathbf{z}) = \nabla_{\boldsymbol{\lambda}} \log q(\mathbf{z};\boldsymbol{\lambda})$, and note again that its expected value is zero. With this function, each component n of the gradient in (1) can be rewritten as $\mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})} \left[f_n(\mathbf{z}) - a_n h_n(\mathbf{z})\right]$ where a_n is a constant and $f(\mathbf{z})$ is defined in (2). (Here, $f_n(\mathbf{z})$ and $h_n(\mathbf{z})$ denote the n-th component of $f(\mathbf{z})$ and $h(\mathbf{z})$, respectively.) We can set each element a_n to minimize the variance of the Monte Carlo estimates of this expectation,

$$a_n = \frac{\mathrm{Cov}(f_n(\mathbf{z}), h_n(\mathbf{z}))}{\mathrm{Var}(h_n(\mathbf{z}))}. \tag{8}$$

[1] A conditionally conjugate model is a model for which all the complete conditionals (i.e., the posterior distribution of each hidden variable conditioned on the observations and the rest of hidden variables) are in the same exponential family as the prior.

In BBVI, a separate set of samples from $q(\mathbf{z};\boldsymbol{\lambda})$ is used to estimate a_n (otherwise, the estimator would be biased).

Rao-Blackwellization. Rao-Blackwellization (Casella and Robert, 1996) reduces the variance of a random variable by replacing it with its conditional expectation, given a subset of other variables. In BBVI, each component of the gradient is Rao-Blackwellized with respect to variables outside of the Markov blanket of the involved hidden variable. More precisely, assume a mean-field[2] variational distribution $q(\mathbf{z};\boldsymbol{\lambda}) = \prod_n q(z_n;\lambda_n)$. We can equivalently rewrite the expectation of each element in Eq. 1 as

$$\nabla_{\lambda_n} \mathcal{L} = \mathbb{E}_{q(\mathbf{z}_{(n)};\boldsymbol{\lambda}_{(n)})}\big[\nabla_{\lambda_n} \log q(z_n;\lambda_n) \times \big(\log p_n(\mathbf{x}, \mathbf{z}_{(n)}) - \log q(z_n;\lambda_n)\big)\big], \tag{9}$$

where $\mathbf{z}_{(n)}$ denotes the variable z_n together with all latent variables in its Markov blanket, $q(\mathbf{z}_{(n)};\boldsymbol{\lambda}_{(n)})$ denotes the variational distribution on $\mathbf{z}_{(n)}$, and $\log p_n(\mathbf{x}, \mathbf{z}_{(n)})$ contains all terms of the log-joint distribution that depend on $\mathbf{z}_{(n)}$. The Monte Carlo estimate based on the Rao-Blackwellized expectation has significantly smaller variance than the estimator driven by Eq. 1.

3 OVERDISPERSED BLACK-BOX VARIATIONAL INFERENCE

We have described BBVI and its two strategies for reducing the variance of the noisy gradient. We now describe O-BBVI, a method for further reducing the variance. The main idea is to use importance sampling (Robert and Casella, 2005) to estimate the gradient. We first describe O-BBVI and the proposal distribution it uses. We then show that this reduces variance, discuss several important implementation details, and present the full algorithm.

O-BBVI does not sample from the variational distribution $q(\mathbf{z};\boldsymbol{\lambda})$ to estimate the expectation $\mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})} \left[f(\mathbf{z})\right]$. Rather, it takes samples from a proposal distribution $r(\mathbf{z};\boldsymbol{\lambda},\tau)$ and constructs estimates of the gradient in Eq. 3, where the importance weights are $w(\mathbf{z}) = q(\mathbf{z};\boldsymbol{\lambda})/r(\mathbf{z};\boldsymbol{\lambda},\tau)$. This guarantees that the resulting estimator is unbiased. The proposal distribution involves the current setting of the variational parameters $\boldsymbol{\lambda}$ and an additional parameter τ.

The optimal proposal. The particular proposal that O-BBVI uses is inspired by a result from the importance sampling literature (Robert and Casella, 2005; Owen, 2013). This result states that the optimal proposal distribution, which minimizes the variance of the estimator, is *not* the variational distribution $q(\mathbf{z};\boldsymbol{\lambda})$. Rather, the optimal pro-

[2] A structured variational approach is also amenable to Rao-Blackwellization, but we assume a fully factorized variational distribution for simplicity.

posal is

$$r_n^\star(\mathbf{z}) \propto q(\mathbf{z};\boldsymbol{\lambda})\,|f_n(\mathbf{z})|, \tag{10}$$

for each component n of the gradient. (Recall that $f(\mathbf{z})$ is a vector of the same length as $\boldsymbol{\lambda}$.)

While interesting, the optimal proposal distribution is not tractable in general—it involves normalizing a complex product—and is not "black box" in the sense that it depends on the model via $f(\mathbf{z})$. In O-BBVI, we build an alternative proposal based on overdispersed exponential families (Jørgensen, 1987). We will argue that this proposal is closer to the (intractable) optimal $r^*(\mathbf{z})$ than the variational distribution $q(\mathbf{z};\boldsymbol{\lambda})$, and that it is still practical in the context of stochastic optimization of the variational objective. Note that approximating the optimal proposal in stochastic optimization was also explored in Bouchard et al. (2015).

The overdispersed proposal. Our motivation for using overdispersed exponential families is that the optimal distribution of Eq. 10 assigns higher probability density to the tails of $q(\mathbf{z};\boldsymbol{\lambda})$. There are two reasons for this fact. First, consider settings of the variational parameters where the variational distribution is a poor fit to the posterior. For these parameters, there are values of $\mathbf{z}$ for which the posterior is high but the variational distribution is small. While the optimal proposal would sample configurations of $\mathbf{z}$ for which $f_n(\mathbf{z})$ is large, these realizations are in the tails of the variational distribution.

The second reason has to do with the score function. The score function $h_n(\mathbf{z})$ vanishes for values of $\mathbf{z}$ for which the n-th sufficient statistic $t_n(\mathbf{z})$ equals its expected value, and this pushes probability mass (in the optimal proposal) to the tails of $q(\mathbf{z};\boldsymbol{\lambda})$. To see this, recall the exponential family form of the variational distribution given in Eq. 4. For any exponential family distribution, the score function is $h_n(\mathbf{z}) = t_n(\mathbf{z}) - \mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})}[t_n(\mathbf{z})]$. (This result follows from simple properties of exponential families.[3]) For values of $\mathbf{z}$ for which $t_n(\mathbf{z})$ is close to its expectation, $h_n(\mathbf{z})$ becomes very close to zero. This zeros out $f_n(\mathbf{z})$ in Eq. 10, which pushes mass to other parts of $q(\mathbf{z};\boldsymbol{\lambda})$. As an example, in the case where $q(\mathbf{z};\boldsymbol{\lambda})$ is a Gaussian distribution, the optimal proposal distribution places zero mass on the mean of that Gaussian and hence more probability mass on its tails.

Thus, we design a proposal distribution $r(\mathbf{z};\boldsymbol{\lambda},\tau)$ that assigns higher mass to the tails of $q(\mathbf{z};\boldsymbol{\lambda})$. Specifically, we use an overdispersed distribution in the same exponential family as $q(\mathbf{z};\boldsymbol{\lambda})$. The proposal is

$$r(\mathbf{z};\boldsymbol{\lambda},\tau) = g(\mathbf{z},\tau)\exp\left\{\frac{\boldsymbol{\lambda}^\top t(\mathbf{z}) - A(\boldsymbol{\lambda})}{\tau}\right\}, \tag{11}$$

[3]The gradient of the log-normalizer (with respect to the natural parameters) equals the first-order moment of the sufficient statistics, i.e., $\nabla_{\boldsymbol{\lambda}} A(\boldsymbol{\lambda}) = \mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})}[t(\mathbf{z})]$.

where $\tau \geq 1$ is the dispersion coefficient of the overdispersed distribution (Jørgensen, 1987). Hence, the O-BBVI estimator of the gradient can be expressed as

$$\widehat{\nabla}^{\text{O-BB}}_{\boldsymbol{\lambda}}\mathcal{L} = \frac{1}{S}\sum_s f(\mathbf{z}^{(s)})\frac{q(\mathbf{z}^{(s)};\boldsymbol{\lambda})}{r(\mathbf{z}^{(s)})}, \quad \mathbf{z}^{(s)} \overset{\text{iid}}{\sim} r(\mathbf{z};\boldsymbol{\lambda},\tau), \tag{12}$$

where S is the number of samples of the Monte Carlo approximation.

This choice of $r(\mathbf{z};\boldsymbol{\lambda},\tau)$ has several desired properties for a proposal distribution. First, it is easy to sample from, since for fixed values of τ it belongs to the same exponential family as $q(\mathbf{z};\boldsymbol{\lambda})$. Second, as for the optimal proposal, it is adaptive, since it explicitly depends on the parameters $\boldsymbol{\lambda}$ which we are optimizing. Finally, by definition, it assigns higher mass to the tails of $q(\mathbf{z};\boldsymbol{\lambda})$, which was our motivation for choosing it.

The dispersion coefficient τ can be itself adaptive to better match the optimal proposal at each iteration of the variational optimization procedure. We put forward a method to update the value of τ in Section 3.2.

Note that our approach differs from importance weighted autoencoders (Burda et al., 2016), which also make use of importance sampling but with the goal of deriving a tighter log-likelihood lower bound in the context of the variational autoencoder (Kingma and Welling, 2014). In contrast, we use importance sampling to reduce the variance of the estimator of the gradient.

3.1 Variance reduction

Here, we compare the variance of the O-BBVI estimator $\widehat{\nabla}^{\text{O-BB}}_{\lambda}\mathcal{L}$ given in Eq. 12 with the variance of the original BBVI estimator $\widehat{\nabla}^{\text{BB}}_{\lambda}\mathcal{L}$, which samples from $q(\mathbf{z};\boldsymbol{\lambda})$:

$$\widehat{\nabla}^{\text{BB}}_{\boldsymbol{\lambda}}\mathcal{L} = \frac{1}{S}\sum_s f(\mathbf{z}^{(s)}), \qquad \mathbf{z}^{(s)} \overset{\text{iid}}{\sim} q(\mathbf{z};\boldsymbol{\lambda}). \tag{13}$$

After some algebra, we can express the variance of the BBVI estimator as

$$\mathbb{V}\left[\widehat{\nabla}^{\text{BB}}_{\boldsymbol{\lambda}}\mathcal{L}\right] = \frac{1}{S}\mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})}\left[f^2(\mathbf{z})\right] - \frac{1}{S}(\nabla_{\boldsymbol{\lambda}}\mathcal{L})^2, \tag{14}$$

and we can also express the variance of the O-BBVI estimator in terms of an expectation with respect to the variational distribution as

$$\begin{aligned}\mathbb{V}\left[\widehat{\nabla}^{\text{O-BB}}_{\boldsymbol{\lambda}}\mathcal{L}\right] &= \frac{1}{S}\mathbb{E}_{r(\mathbf{z};\boldsymbol{\lambda},\tau)}\left[f^2(\mathbf{z})\frac{q^2(\mathbf{z};\boldsymbol{\lambda})}{r^2(\mathbf{z};\boldsymbol{\lambda},\tau)}\right] - \frac{1}{S}(\nabla_{\boldsymbol{\lambda}}\mathcal{L})^2 \\ &= \frac{1}{S}\mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})}\left[f^2(\mathbf{z})\frac{q(\mathbf{z};\boldsymbol{\lambda})}{r(\mathbf{z};\boldsymbol{\lambda},\tau)}\right] - \frac{1}{S}(\nabla_{\boldsymbol{\lambda}}\mathcal{L})^2.\end{aligned} \tag{15}$$

Variance reduction for the O-BBVI approach is achieved when $\mathbb{V}\left[\widehat{\nabla}^{\text{O-BB}}_{\boldsymbol{\lambda}}\mathcal{L}\right] \leq \mathbb{V}\left[\widehat{\nabla}^{\text{BB}}_{\boldsymbol{\lambda}}\mathcal{L}\right]$ or, equivalently,

$$\mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})}\left[f^2(\mathbf{z})\frac{q(\mathbf{z};\boldsymbol{\lambda})}{r(\mathbf{z};\boldsymbol{\lambda},\tau)}\right] \leq \mathbb{E}_{q(\mathbf{z};\boldsymbol{\lambda})}\left[f^2(\mathbf{z})\right]. \tag{16}$$

This inequality is trivially satisfied when we set $r(\mathbf{z})$ to the optimal proposal distribution, $r^\star(\mathbf{z})$. However, it is intractable to compute in general; moreover, it depends on the model through $f(\mathbf{z})$. While the bound in Eq. 16 results intractable, it gives us some intuition on why the use of an overdispersed proposal distribution can reduce the variance, since $r(\mathbf{z};\boldsymbol{\lambda},\tau)$ will be larger than $q(\mathbf{z};\boldsymbol{\lambda})$ for those values of $\mathbf{z}$ for which the product $q(\mathbf{z};\boldsymbol{\lambda})f^2(\mathbf{z})$ is highest, i.e., in the tails of $q(\mathbf{z};\boldsymbol{\lambda})$. Our experimental results in Section 4 demonstrate that the variance is effectively reduced when we use our O-BBVI.

3.2 Implementation

We now discuss several extensions of O-BBVI that make it more suitable for real applications.

High dimensionality. Previously, we defined the proposal distribution $r(\mathbf{z};\boldsymbol{\lambda},\tau)$ as an overdispersed version of the variational distribution $q(\mathbf{z};\boldsymbol{\lambda})$. However, importance sampling is known to fail when the dimensionality of the hidden space is moderately high, due to the high resulting variance of the importance weights $w(\mathbf{z}) = q(\mathbf{z};\boldsymbol{\lambda})/r(\mathbf{z};\boldsymbol{\lambda},\tau)$. To address this, we rely on the fact that hidden variable z_n is the variable with the highest influence on the estimator of the n-th component of the gradient. We exploit this idea, which was also considered by Titsias and Lázaro-Gredilla (2015) in their algorithm based on local expectations.

More precisely, for the variational parameters of variable z_n, we first write the gradient as

$$\begin{aligned}\nabla_{\lambda_n}\mathcal{L} &= \mathbb{E}_{q(z_n;\lambda_n)}\left[\mathbb{E}_{q(\mathbf{z}_{\neg n};\boldsymbol{\lambda}_{\neg n})}\left[f_n(\mathbf{z})\right]\right] \\ &= \mathbb{E}_{r(z_n;\lambda_n,\tau_n)}\left[w(z_n)\mathbb{E}_{q(\mathbf{z}_{\neg n};\boldsymbol{\lambda}_{\neg n})}\left[f_n(\mathbf{z})\right]\right],\end{aligned} \tag{17}$$

where $r(z_n;\lambda_n,\tau_n)$ is the overdispersed version of $q(z_n;\lambda_n)$ with dispersion coefficient τ_n, $\mathbf{z}_{\neg n}$ denotes all hidden variables in the model except z_n, and similarly for $\boldsymbol{\lambda}_{\neg n}$. Thus, the corresponding importance weights in (17) for each component of the gradient depend only on variable z_n, i.e.,

$$w(z_n) = \frac{q(z_n;\lambda_n)}{r(z_n;\lambda_n,\tau_n)}. \tag{18}$$

We use a single sample from $q(\mathbf{z}_{\neg n};\boldsymbol{\lambda}_{\neg n})$ to estimate the inner expectation in (17), and S samples of z_n from $r(z_n;\lambda_n,\tau_n)$ to estimate the outer expectation.

Adaptation of the dispersion coefficients. Our algorithm requires setting the value of the dispersion parameters τ_n; we would like to automate this procedure. Here, we develop a method to learn these coefficients during optimization by minimizing the variance of the estimator. More precisely, we introduce stochastic gradient descent steps for τ_n that minimize the variance. The exact derivative of the (negative) variance with respect to τ_n is

$$\begin{aligned}-\frac{\partial\mathbb{V}\left[\widehat{\nabla}^{\text{O-BB}}_{\lambda_n}\mathcal{L}\right]}{\partial\tau_n} &= \frac{1}{S}\mathbb{E}_{r(z_n;\lambda_n,\tau_n)}\Bigg[\mathbb{E}_{q(\mathbf{z}_{\neg n};\boldsymbol{\lambda}_{\neg n})}\left[f_n(\mathbf{z})\right]^2 \\ &\quad\times w^2(z_n)\frac{\partial\log r(z_n;\lambda_n,\tau_n)}{\partial\tau_n}\Bigg],\end{aligned} \tag{19}$$

where we have applied the log-derivative trick once again, as well as the extension to high dimensionality detailed above. Now a Monte Carlo estimate of this derivative can be obtained by using the same set of S samples used in the update of λ_n. The resulting procedure is fast, with little extra overhead, since both $f_n(\mathbf{z})$ and $w(z_n)$ have been pre-computed.

Thus, we perform gradient steps of the form

$$\tau_n^{(t)} = \tau_n^{(t-1)} - \alpha_n\frac{\partial\mathbb{V}\left[\widehat{\nabla}^{\text{O-BB}}_{\lambda_n}\mathcal{L}\right]}{\partial\tau_n}, \tag{20}$$

where τ_n is constrained as $\tau_n \geq 1$ and the derivatives are estimated via Monte Carlo approximation. Since the derivatives in Eq. 20 can be several orders of magnitude greater than τ_n, we opt for a simple approach to choose an appropriate step size α_n. In particular, we ignore the magnitude of the derivative in (20) and take a small gradient step in the direction given by its sign. Note that we do not need to satisfy the Robbins-Monro conditions here (Robbins and Monro, 1951), because the adaptation of τ_n only defines the proposal distribution and it is not part of the original stochastic optimization procedure.

Eq. 20 can still be applied even if λ_n is a vector; it only requires replacing the derivative of the variance with the summation of the derivatives for all components of λ_n.

Multiple importance sampling. It may be more stable (in terms of the variance of the importance weights) to consider a set of J dispersion coefficients, $\tau_{n1},\ldots,\tau_{nJ}$, instead of a single coefficient τ_n. We propose to use a mixture with equal weights to build the proposal as follows:

$$r(z_n;\lambda_n,\tau_{n1},\ldots,\tau_{nJ}) = \frac{1}{J}\sum_{j=1}^{J} r(z_n;\lambda_n,\tau_{nj}), \tag{21}$$

where each term in the mixture is given by $r(z_n;\lambda_n,\tau_{nj}) = g(z_n,\tau_{nj})\exp\left\{\frac{\lambda_n^\top t(z_n)-A(\lambda_n)}{\tau_{nj}}\right\}$. In the importance sampling literature, this is known as multiple importance sampling (MIS), as multiple proposals are used (Veach and Guibas, 1995). Within the MIS methods, we opt for full deterministic multiple importance sampling (DMIS) because it is the approach that presents lowest variance (Hesterberg, 1995; Owen and Zhou, 2000; Elvira et al., 2015). In DMIS, the number of samples S of the Monte Carlo estimator must

be an integer multiple of the number of mixture components J, and S/J samples are deterministically assigned to each proposal $r(z_n; \lambda_n, \tau_{nj})$. However, the importance weights are obtained as if the samples had been actually drawn from the mixture, i.e.,

$$w(z_n) = \frac{q(z_n; \lambda_n)}{\frac{1}{J}\sum_{j=1}^{J} r(z_n; \lambda_n, \tau_{nj})}. \tag{22}$$

This choice of the importance weights yields an unbiased estimator with smaller variance than the standard MIS approach (Owen and Zhou, 2000; Elvira et al., 2015).

In the experiments in Section 4 we investigate the performance of two-component proposal distributions, where $J = 2$, and compare it against our initial algorithm that uses a unique proposal, which corresponds to $J = 1$. We have also conducted some additional experiments (not shown in the paper) with mixtures with higher number of components, with no significant improvements.

3.3 Full algorithm

We now present our full algorithm for O-BBVI. It makes use of control variates, Rao-Blackwellization, and overdispersed importance sampling with adaptation of the dispersion coefficients. At each iteration, we draw a single sample $\mathbf{z}^{(0)}$ from the variational distribution, as well as S samples $z_n^{(s)}$ from the overdispersed proposal for each n (using DMIS in this step). We obtain the score function as

$$h_n(z_n^{(s)}) = \nabla_{\lambda_n} \log q(z_n^{(s)}; \lambda_n), \tag{23}$$

and the argument of the expectation in (9) as

$$f_n(\mathbf{z}^{(s)}) = h_n(z_n^{(s)})(\log p_n(\mathbf{x}, z_n^{(s)}, \mathbf{z}_{\neg n}^{(0)}) - \log q(z_n^{(s)}; \lambda_n)), \tag{24}$$

where p_n indicates that we use Rao-Blackwellization. Finally, the estimator of the gradient is obtained as

$$\widehat{\nabla}_{\lambda_n} \mathcal{L} = \frac{1}{S}\sum_{s}\left(f_n^{\mathrm{w}}(\mathbf{z}^{(s)}) - a_n h_n^{\mathrm{w}}(z_n^{(s)})\right), \tag{25}$$

where the superscript "w" stands for "weighted," i.e.,

$$f_n^{\mathrm{w}}(\mathbf{z}^{(s)}) = w(z_n^{(s)}) f_n(\mathbf{z}^{(s)}), \tag{26}$$

$$h_n^{\mathrm{w}}(z_n^{(s)}) = w(z_n^{(s)}) h_n(z_n^{(s)}). \tag{27}$$

Following Eq. 8, we use a separate set of samples to estimate the optimal a_n as

$$a_n = \frac{\widehat{\mathrm{Cov}}(f_n^{\mathrm{w}}, h_n^{\mathrm{w}})}{\widehat{\mathrm{Var}}(h_n^{\mathrm{w}})}. \tag{28}$$

We use AdaGrad (Duchi et al., 2011) to obtain adaptive learning rates that ensure convergence of the stochastic optimization procedure, although other schedules can be used instead as long as they satisfy the standard Robbins-Monro conditions (Robbins and Monro, 1951). In AdaGrad, the learning rate is obtained as

$$\boldsymbol{\rho}_t = \eta \ \mathrm{diag}\left(\mathbf{G}_t\right)^{-1/2}, \tag{29}$$

where $\mathbf{G}_t$ is a matrix that contains the sum across the first t iterations of the outer products of the gradient, and η is a constant. Thus, the stochastic gradient step is given by

$$\boldsymbol{\lambda}^{(t)} = \boldsymbol{\lambda}^{(t-1)} + \boldsymbol{\rho}_t \circ \widehat{\nabla}_{\boldsymbol{\lambda}} \mathcal{L}, \tag{30}$$

where '$\circ$' denotes the element-wise (Hadamard) product. Algorithm 1 summarizes the full procedure.

Algorithm 1: Overdispersed black-box variational inference (O-BBVI)

input : data $\mathbf{x}$, joint distribution $p(\mathbf{x}, \mathbf{z})$, mean-field variational family $q(\mathbf{z}; \boldsymbol{\lambda})$
output: variational parameters $\boldsymbol{\lambda}$
Initialize $\boldsymbol{\lambda}$
Initialize the dispersion coefficients τ_{nj}
while *algorithm has not converged* **do**
  `/* draw samples */`
  Draw a single sample $\mathbf{z}^{(0)} \sim q(\mathbf{z}; \boldsymbol{\lambda})$
  for $n = 1$ to N **do**
    Draw S samples $z_n^{(s)} \sim r(z_n; \lambda_n, \{\tau_{nj}\})$ (DMIS)
    Compute the importance weights $w(z_n^{(s)})$ (Eq. 22)
  end
  `/* estimate gradient */`
  for $n = 1$ to N **do**
    For each sample s, compute $h_n(z_n^{(s)})$ (Eq. 23)
    For each sample s, compute $f_n(\mathbf{z}^{(s)})$ (Eq. 24)
    Compute the weighted $f_n^{\mathrm{w}}(\mathbf{z}^{(s)})$ (Eq. 26)
    Compute the weighted $h_n^{\mathrm{w}}(z_n^{(s)})$ (Eq. 27)
    Estimate the optimal a_n (Eq. 28)
    Estimate the gradient $\widehat{\nabla}_{\lambda_n} \mathcal{L}$ (Eq. 25)
  end
  `/* update dispersion coefficients */`
  for $n = 1$ to N **do**
    Estimate the derivatives $\frac{\partial \mathbb{V}\left[\nabla_{\lambda_n} \mathcal{L}\right]}{\partial \tau_{nj}}$ (Eq. 19)
    Take a gradient step for τ_{nj} (Eq. 20)
  end
  `/* take gradient step */`
  Set the step size $\boldsymbol{\rho}_t$ (Eq. 29)
  Take a gradient step for $\boldsymbol{\lambda}$ (Eq. 30)
end

4 EMPIRICAL STUDY

We study our method with two non-conjugate probabilistic models: the gamma-normal time series model (GN-TS) and the Poisson deep exponential family (DEF). We

found that overdispersed black-box variational inference (O-BBVI) reduces the variance of the black-box variational inference (BBVI) estimator and leads to faster convergence.

4.1 Description of the experiments

Models description and datasets. The GN-TS model (Ranganath et al., 2014) is a non-conjugate state-space model for sequential data that was used to showcase BBVI. The model is described by

$$\begin{aligned} w_{kd} &\sim \mathcal{N}(0, \sigma_w^2), \\ o_{nd} &\sim \mathcal{N}(0, \sigma_o^2), \\ z_{n1k} &\sim \text{GammaE}(\sigma_z, \sigma_z), \\ z_{ntk} &\sim \text{GammaE}(z_{n(t-1)k}, \sigma_z), \\ x_{ndt} &\sim \mathcal{N}\left(o_{nd} + \sum_k z_{ntk} w_{kd}, \sigma_x^2\right). \end{aligned} \tag{31}$$

The indices n, t, d and k denote observations, time instants, observation dimensions, and latent factors, respectively. The distribution GammaE denotes the expectation/variance parameterization of the gamma distribution. The model explains each datapoint x_{ndt} with a latent factor model. For each time instant t, the mean of x_{ndt} depends on the inner product $\sum_k z_{ntk} w_{kd}$, where z_{ntk} varies smoothly across time. The variables o_{nd} are an intercept that capture the baseline in the observations.

We set the hyperparameters to be $\sigma_w^2 = 1, \sigma_o^2 = 1, \sigma_z = 1$, and $\sigma_x^2 = 0.01$. We use a synthetic dataset of $N = 900$ time sequences of length $T = 30$ and dimensionality $D = 20$. We use $K = 30$ latent factors, leading to $828,600$ hidden variables.

The Poisson DEF (Ranganath et al., 2015) is a multilayered latent variable model of discrete data, such as text. The model is described by

$$\begin{aligned} w_{kv}^{(0)} &\sim \text{Gamma}(\alpha_w, \beta_w), \\ w_{kk'}^{(\ell)} &\sim \text{Gamma}(\alpha_w, \beta_w), \\ z_{dk}^{(L)} &\sim \text{Poisson}(\lambda_z), \\ z_{dk}^{(\ell)} &\sim \text{Poisson}\left(\sum_{k'} z_{dk'}^{(\ell+1)} w_{k'k}^{(\ell)}\right), \\ x_{dv} &\sim \text{Poisson}\left(\sum_{k'} z_{dk'}^{(1)} w_{k'v}^{(0)}\right). \end{aligned} \tag{32}$$

The indices d, v, k and ℓ denote documents, vocabulary words, latent factors, and hidden layers, respectively. This model captures a hierarchy of dependencies between latent variables similar to the hidden structure in deep neural networks. In detail, the number of times that word v appears in document d is x_{dv}. It has a Poisson distribution with rate given by an inner product of gamma-distributed weights and Poisson-distributed hidden variables from layer 1. The Poisson-distributed hidden variables depend, in turn, on another set of weights and another layer of hidden Poisson-distributed variables. This structure repeats for a specified number of layers.

We set the prior shape and rate as $\alpha_w = 0.1$ and $\beta_w = 0.3$, and the prior mean for the top level of the Poisson DEF as $\lambda_z = 0.1$. We use $L = 3$ layers with $K = 50$ latent factors each. We model the papers at the Neural Information Processing Systems (NIPS) 2011 conference. This is a data set with $D = 305$ documents, $612,508$ words, and $V = 5715$ vocabulary words (after removing stop words). This leads to a model with $336,500$ hidden variables.

Evaluation. We compare O-BBVI with BBVI (Ranganath et al., 2014). For a fair comparison, we use the same number of samples in both methods and estimate the inner expectation in Eq. 17 with only one sample. For the outer expectation, we use 8 samples to estimate the gradient itself and 8 separate samples to estimate the optimal coefficient a_n for the control variates. For BBVI, we also doubled the number of samples to $16 + 16$; this is marked as "BBVI ($\times 2$)" in the plots.

For O-BBVI, we study both a single proposal and a mixture proposal with two components, respectively labeled as "O-BBVI (single proposal)" and "O-BBVI (mixture)." For the latter, we fix the dispersion coefficients $\tau_{n1} = 1$ for all n, and we run stochastic gradient descent steps for τ_{n2}. See the Supplement for some figures showing the evolution of the dispersion coefficient.

At each iteration (and for each method) we evaluate several quantities: the evidence lower bound (ELBO), the averaged sample variance of the estimator of the gradient, and a model-specific performance metric on the test set. The estimation of the ELBO is based on a single sample of the variational distribution $q(\mathbf{z}; \boldsymbol{\lambda})$ for all methods. For the GN-TS model, we compute the average log-likelihood (up to a constant term) on the test set, which is generated with one additional time instant in all sequences. For the Poisson DEF, we compute the average held-out perplexity,

$$\exp\left(\frac{-\sum_d \sum_{w \in \text{doc}(d)} \log p(w \mid \#\text{held out in } d)}{\#\text{held out words}}\right), \tag{33}$$

where the held-out data contains 25% randomly selected words of all documents.

Experimental setup. For each method, we initialize the variational parameters to the same point and run each algorithm with a fixed computational budget (of CPU time).

We use AdaGrad (Duchi et al., 2011) for the learning rate. We set the parameter η in Eq. 29 to $\eta = 0.5$ for the GN-TS model and $\eta = 1$ for the Poisson DEF. When optimizing the O-BBVI dispersion coefficients τ_n, we take steps of length 0.1 in the direction of the (negative) gradient. We initialize the dispersion coefficients as $\tau_n = 2$ for the single

proposal and $\tau_{n2} = 3$ for the two-component mixture.

We parameterize the normal distribution in terms of its mean and variance, the gamma in terms of its shape and mean, and the Poisson in terms of its mean parameter. In order to avoid constrained optimization, we apply the transformation $\lambda' = \log(\exp(\lambda) - 1)$ to those variational parameters that are constrained to be positive and take stochastic gradient steps with respect to λ'.

Overdispersed exponential families. For a fixed dispersion coefficient τ, the overdispersed exponential family of the Gaussian distribution with mean μ and variance σ^2 is a Gaussian distribution with mean μ and variance $\tau\sigma^2$. The overdispersed gamma distribution with shape s and rate r is given by a new gamma distribution with shape $\frac{s+\tau-1}{\tau}$ and rate $\frac{r}{\tau}$. The overdispersed Poisson(λ) distribution is a Poisson($\lambda^{1/\tau}$) distribution.

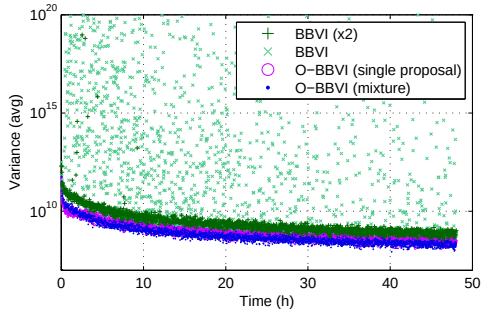
4.2 Results

Figures 1 and 2 show the evolution of the ELBO, the predictive performance, and the average sample variance of the estimator for both models and all methods. We plot these metrics as a function of running time, and each method is run with the same computational budget.

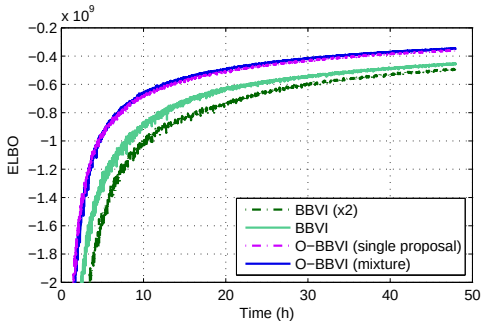
For the GN-TS model, Figure 1a shows that the variance of O-BBVI is significantly lower than BBVI and BBVI with twice the number of samples. Additionally, Figures 1b and 1c show that O-BBVI outperforms vanilla BBVI in terms of both ELBO and held-out likelihood. According to these figures, using a single or mixture proposal does not seem to significantly affect performance. In these plots, we can also see that BBVI ($\times 2$) converges slower than BBVI; this is because the x-axis represents running time instead of iterations. (When the number of samples increases, the convergence is faster but each iteration takes more time.)

The results on the Poisson DEF are similar (Figure 2). Figure 2a shows the average sample variance of the estimator; again, O-BBVI outperforms both BBVI algorithms. Figures 2b and 2c show the evolution of the ELBO and the held-out perplexity, respectively, where O-BBVI also outperforms BBVI. Here, the two-component mixture proposal performs slightly better than the single proposal. This is consistent with Figure 2a, which indicates that the mixture proposal gives more stable estimates.

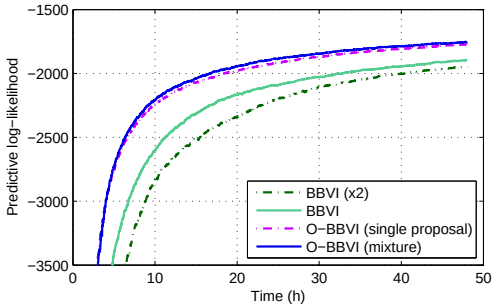
Finally, for the GN-TS model only, we also apply the local expectations algorithm of Titsias and Lázaro-Gredilla (2015), which relies on exact or numerical integration to reduce the variance of the estimator. We form noisy gradients using numerical quadratures for the Gaussian random variables and standard BBVI for the gamma variables (these results are not plotted in the paper). We found that local expectations accurately approximate the gradient for the Gaussian distributions. It converges slightly faster at the beginning of the run, although O-BBVI quickly reaches the same performance. (We conjecture that it is faster because the local expectations algorithm does not require the use of control variates. This saves evaluations of the log-joint probability of the model and thus it can run more iterations in the same period of time.)



(a) Averaged sample variance of the estimator.



(b) Traceplot of the ELBO.

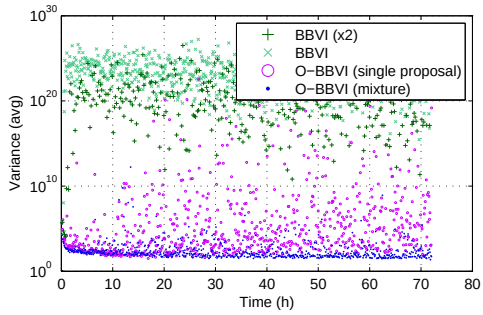


(c) Predictive performance (higher is better).

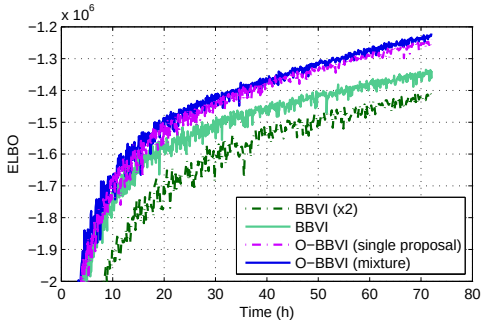
Figure 1: Results for the GN-TS model. Both versions of O-BBVI converge faster than BBVI.

However, we emphasize that O-BBVI is a more general algorithm than local expectations. The local expectations of Titsias and Lázaro-Gredilla (2015) are only available for discrete distributions with finite support and for continuous distributions for which numerical quadratures are accurate (such as Gaussian distributions). They fail to approximate the expectations for other exponential family distributions (e.g., gamma,[4] Poisson, and others). For example, they cannot handle the Poisson DEF.

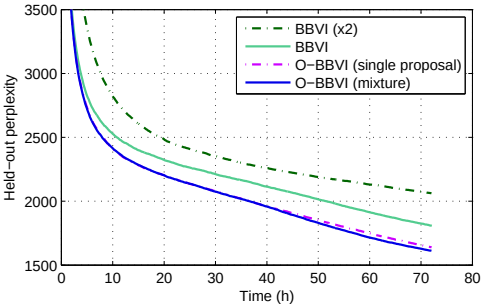
[4]Although the univariate gamma distribution is amenable to numerical integration, we have found that the approximation is not accurate when the shape parameter of the gamma distribution is below 1, due to the singularity at 0.



(a) Averaged sample variance of the estimator.



(b) Traceplot of the ELBO.



(c) Predictive performance (lower is better).

Figure 2: Results for the Poisson DEF model. Both versions of O-BBVI converge faster than BBVI.

5 CONCLUSIONS

We have developed overdispersed black-box variational inference (O-BBVI), a method that relies on importance sampling to reduce the variance of the stochastic gradients in black-box variational inference (BBVI). O-BBVI uses an importance sampling proposal distribution that has heavier tails than the actual variational distribution. In particular, we choose the proposal as an overdispersed distribution in the same exponential family as the variational distribution. Like BBVI, our approach is amenable to mean field or structured variational inference, as well as variational models (Ranganath et al., 2016; Tran et al., 2016).

We have studied the performance of our method on two complex probabilistic models. Our results show that BBVI effectively benefits from the use of overdispersed importance sampling, and O-BBVI leads to faster convergence in the resulting stochastic optimization procedure.

There are several avenues for future work. First, we can explore other proposal distributions to provide a better fit to the optimal ones while still maintaining computational efficiency. Further theoretical research on the bound in Eq. 16 may be helpful for that purpose. Second, we can apply quasi-Monte Carlo methods to further decrease the sampling variance, as already suggested by Ranganath et al. (2014). Finally, we can combine the reparameterization trick with overdispersed proposals to explore whether variance is further reduced.

Acknowledgements

This work is supported by IIS-1247664, ONR N00014-11-1-0651, DARPA FA8750-14-2-0009, DARPA N66001-15-C-4032, Adobe, the John Templeton Foundation, and the Sloan Foundation. The authors thank Rajesh Ranganath and Guillaume Bouchard for useful discussions.

References

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

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

Bouchard, G., Trouillon, T., Perez, J., and Gaidon, A. (2015). Online learning to sample. *arXiv:1506.09016*.

Burda, Y., Grosse, R., and Salakhutdinov, R. (2016). Importance weighted autoencoders. In *International Conference on Learning Representations*.

Carbonetto, P., King, M., and Hamze, F. (2009). A stochastic approximation method for inference in probabilistic graphical models. In *Advances in Neural Information Processing Systems*.

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

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

Elvira, V., Martino, L., Luengo, D., and Bugallo, M. F. (2015). Generalized multiple importance sampling. *arXiv:1511.03095*.

Ghahramani, Z. and Beal, M. J. (2001). Propagation algorithms for variational Bayesian learning. In *Advances in Neural Information Processing Systems*.

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

Gu, S., Levine, S., Sutskever, I., and Mnih, A. (2016). MuProp: Unbiased backpropagation for stochastic neural networks. In *International Conference on Learning Representations*.

Hesterberg, T. (1995). Weighted average importance sampling and defensive mixture distributions. *Technometrics*, 37(2):185–194.

Hinton, G., Dayan, P., Frey, B. J., and Neal, R. M. (1995). The wake-sleep algorithm for unsupervised neural networks. *Science*, 268(5214):1158–1161.

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

Jørgensen, B. (1987). Exponential dispersion models. *Journal of the Royal Statistical Society, Series B (Methodological)*, 49(2):127–162.

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

Kleijnen, J. P. C. and Rubinstein, R. Y. (1996). Optimization and sensitivity analysis of computer simulation models by the score function method. Technical report, Tilburg University, School of Economics and Management.

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

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

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. MIT Press.

Neal, R. (1992). Connectionist learning of belief networks. *Artificial Intelligence*, 56(1):71–113.

Owen, A. and Zhou, Y. (2000). Safe and effective importance sampling. *Journal of the American Statistical Association*, 95(449):135—143.

Owen, A. B. (2013). Monte Carlo theory, methods and examples. Book in preparation.

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

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

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

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

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

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

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407.

Robert, C. P. and Casella, G. (2005). *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Ross, S. M. (2002). *Simulation*. Elsevier.

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

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

Titsias, M. K. and Lázaro-Gredilla, M. (2015). Local expectation gradients for black box variational inference. In *Advances in Neural Information Processing Systems*.

Tran, D., Ranganath, R., and Blei, D. M. (2016). Variational Gaussian processes. In *International Conference on Learning Representations*.

van de Meent, J.-W., Tolpin, D., Paige, B., and Wood, F. (2016). Black-box policy search with probabilistic programs. In *Artificial Intelligence and Statistics*.

Veach, E. and Guibas, L. (1995). Optimally combining sampling techniques for Monte Carlo rendering. In *ACM SIGGRAPH*.

Wainwright, M. J. and Jordan, M. I. (2008). Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305.

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

Wingate, D. and Weber, T. (2013). Automated variational inference in probabilistic programming. *arXiv:1301.1299*.