Unbiased Implicit Variational Inference

Michalis K. Titsias
Athens University of Economics and Business

Francisco J. R. Ruiz
University of Cambridge
Columbia University

Abstract

We develop *unbiased implicit variational inference* (UIVI), a method that expands the applicability of variational inference by defining an expressive variational family. UIVI considers an implicit variational distribution obtained in a hierarchical manner using a simple reparameterizable distribution whose variational parameters are defined by arbitrarily flexible deep neural networks. Unlike previous works, UIVI directly optimizes the evidence lower bound (ELBO) rather than an approximation to the ELBO. We demonstrate UIVI on several models, including Bayesian multinomial logistic regression and variational autoencoders, and show that UIVI achieves both tighter ELBO and better predictive performance than existing approaches at a similar computational cost.

1 INTRODUCTION

Variational inference (VI) is an approximate Bayesian inference technique that recasts inference as an optimization problem [Jordan, 1999, Wainwright and Jordan, 2008, Blei et al., 2017]. The goal of VI is to approximate the posterior $p(z \mid x)$ of a given probabilistic model $p(x, z)$, where x denotes the data and z stands for the latent variables. VI posits a parameterized family of distributions $q_\theta(z)$ and then minimizes the Kullback-Leibler (KL) divergence between the approximating distribution $q_\theta(z)$ and the exact posterior $p(z \mid x)$. This minimization is equivalent to maximizing the evidence lower bound (ELBO), which is a function $\mathcal{L}(\theta)$ expressed as an expectation over the variational distribution,

$$\mathcal{L}(\theta) = \mathbb{E}_{q_\theta(z)} \left[\log p(x, z) - \log q_\theta(z) \right]. \tag{1}$$

Thus, VI maximizes Eq. 1, which involves the log-joint $\log p(x, z)$ rather than the intractable posterior.

Classical VI relies on the assumption that the expectations in Eq. 1 are tractable and applies a coordinate-wise ascent algorithm to find θ [Ghahramani and Beal, 2001]. In general, this assumption requires two conditions: the model must be conditionally conjugate (meaning that the conditionals $p(z_n \mid x, z_{\neg n})$ are in the same exponential family as the prior $p(z_n)$ for each latent variable z_n), and the variational family must have a simplified form such as to be factorized across latent variables (mean-field VI).

The above two restrictive conditions when applying VI have motivated several lines of research to expand the use of VI to more complex settings. To address the conjugacy condition on the model, black-box VI methods have been developed, allowing VI to be applied on a broad class of models by using Monte Carlo estimators of the gradient [Carbonetto et al., 2009, Paisley et al., 2012, Ranganath et al., 2014, Kingma and Welling, 2014, Rezende et al., 2014, Titsias and Lázaro-Gredilla, 2014, Kucukelbir et al., 2015, 2017]. To address the simplified (typically mean-field) form of the variational family, more complex variational families have been proposed that incorporate some structure among the latent variables [Jaakkola and Jordan, 1998, Saul and Jordan, 1996, Giordano et al., 2015, Tran et al., 2015, 2016, Ranganath et al., 2016, Han et al., 2016, Maaløe et al., 2016]. See also Zhang et al. [2017] for a review on recent advances on variational inference.

Here, we focus on implicit VI where the variational distribution $q_\theta(z)$ can have arbitrarily flexible forms constructed using neural network mappings. A distribution $q_\theta(z)$ is *implicit* when it is not possible to evaluate its density but it is possible to draw samples from it. One typical way to draw from an implicit distribution in VI is to first sample a noise vector and then push it through a deep neural network [Mohamed and Lakshminarayanan, 2016, Huszár, 2017, Tran et al., 2017, Li and Turner, 2018, Mescheder et al., 2017, Shi et al., 2018a]. Implicit VI expands the variational family making $q_\theta(z)$ more expressive, but computing $\log q_\theta(z)$ in Eq. 1—or its gradient—becomes intractable. To address that, implicit VI typically relies on density ratio estimation. However, density ratio estimation is challenging in high-dimensional settings. To avoid density ratio estimation, Yin and Zhou [2018] proposed semi-implicit variational inference (SIVI), an approach that obtains the variational

Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS) 2019, Naha, Okinawa, Japan. PMLR: Volume 89.

distribution $q_\theta(z)$ by mixing the variational parameter with an implicit distribution. Exploiting this definition of $q_\theta(z)$, SIVI optimizes a sequence of lower (or upper) bounds on the ELBO that eventually converge to Eq. 1.

In this paper, we develop an unbiased estimator of the gradient of the ELBO that avoids density ratio estimation. Our approach builds on SIVI in that we also define the variational distribution by mixing the variational parameter with an implicit distribution. In contrast to SIVI, we propose an unbiased optimization method that directly maximizes the ELBO rather than a bound. We call our method *unbiased implicit variational inference* (UIVI). We show experimentally that UIVI can achieve better ELBO and predictive log-likelihood than SIVI at a similar computational cost.

We develop UIVI using a semi-implicit variational approximation $q_\theta(z) = \int q_\theta(z \,|\, \varepsilon) q(\varepsilon) d\varepsilon$, such that the conditional $q_\theta(z \,|\, \varepsilon)$ is a reparameterizable distribution. The dependence of the conditional $q_\theta(z \,|\, \varepsilon)$ on the random variable ε can be arbitrarily complex. We use a deep neural network parameterized by θ that takes ε as input and outputs the parameters of the conditional $q_\theta(z \,|\, \varepsilon)$. Given ε, the conditional is a "simple" reparameterizable distribution; however marginalizing out ε results in an implicit and more complex distribution $q_\theta(z)$. Exploiting these properties of the variational distribution, UIVI expresses the gradient of the ELBO in Eq. 1 as an expectation, allowing us to construct an unbiased Monte Carlo estimator. The resulting estimator requires samples from the conditional distribution $q_\theta(\varepsilon \,|\, z) \propto q_\theta(z \,|\, \varepsilon) q(\varepsilon)$. We develop a computationally efficient way to draw samples from this conditional using a fast Markov chain Monte Carlo (MCMC) procedure that starts from the stationary distribution. In this way, we avoid the time-consuming burn-in or transient phase that characterizes MCMC methods.

2 UNBIASED IMPLICIT VARIATIONAL INFERENCE

In this section we present unbiased implicit variational inference (UIVI). First, in Section 2.1 we describe how to build the variational distribution, following semi-implicit variational inference (SIVI) [Yin and Zhou, 2018]. Second, in Section 2.2 we show how to form an unbiased estimator of the gradient of the evidence lower bound (ELBO). Finally, in Section 2.3 we put forward the resulting UIVI algorithm and explain how to run it efficiently.

2.1 Semi-Implicit Variational Distribution

To approximate the posterior $p(z \,|\, x)$ of a probabilistic model $p(x, z)$, UIVI uses a semi-implicit variational distribution $q_\theta(z)$ [Yin and Zhou, 2018]. This means that $q_\theta(z)$ is defined in a hierarchical manner with a mixing parameter,

$$\varepsilon \sim q(\varepsilon), \quad z \sim q_\theta(z \,|\, \varepsilon), \tag{2}$$

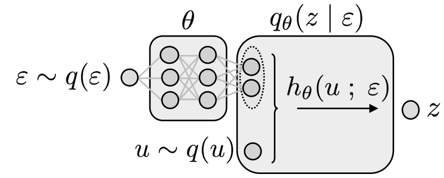


Figure 1. Illustration of the sampling procedure for the implicit variational distribution $q_\theta(z)$. First, a sample $\varepsilon \sim q(\varepsilon)$ is pushed through a neural network with parameters θ (left block). This network outputs the parameters (dotted ellipse) of the conditional distribution $q_\theta(z \,|\, \varepsilon)$. Since the conditional is reparameterizable (right block), to draw a sample z we can first sample $u \sim q(u)$ and then set $z = h_\theta(u\,;\,\varepsilon)$, where $h_\theta(\cdot)$ is an appropriate transformation. The transformation $h_\theta(\cdot)$ depends on ε and θ through the parameters of the conditional. The output $z = h_\theta(u\,;\,\varepsilon)$ is a sample from the variational distribution $q_\theta(z)$.

or equivalently,

$$q_\theta(z) = \int q_\theta(z \,|\, \varepsilon) q(\varepsilon) d\varepsilon. \tag{3}$$

Eqs. 2 and 3 reveal why the resulting variational distribution $q_\theta(z)$ is implicit, as we can obtain samples from it (Eq. 2) but cannot evaluate its density, as the integral in Eq. 3 is intractable.[1]

The dependence of the conditional $q_\theta(z \,|\, \varepsilon)$ on the random variable ε can be arbitrarily complex. In UIVI, its parameters are the output of a deep neural network (parameterized by the variational parameters θ) that takes ε as input.

Assumptions. In UIVI, the conditional $q_\theta(z \,|\, \varepsilon)$ must satisfy two assumptions. First, it must be reparameterizable. That is, to sample from $q_\theta(z \,|\, \varepsilon)$, we can first draw an auxiliary variable u and then set z as a deterministic function $h_\theta(\cdot)$ of the sampled u,

$$u \sim q(u), \quad z = h_\theta(u\,;\,\varepsilon) \quad \equiv \quad z \sim q_\theta(z \,|\, \varepsilon). \tag{4}$$

The transformation $h_\theta(u\,;\,\varepsilon)$ is parameterized by the random variable ε and the variational parameters θ, but the auxiliary distribution $q(u)$ has no parameters. Figure 1 illustrates this construction of the variational distribution.

The second assumption on the conditional $q_\theta(z \,|\, \varepsilon)$ is that it is possible to evaluate the log-density $\log q_\theta(z \,|\, \varepsilon)$ and its gradient with respect to z, $\nabla_z \log q_\theta(z \,|\, \varepsilon)$. This is not a strong assumption; it holds for many reparameterizable distributions (e.g., Gaussian, Laplace, exponential, etc.).

UIVI makes use of these two properties of the conditional

[1] This is similar to a hierarchical variational model [Ranganath et al., 2016]; however we do not assume that the conditional $q_\theta(z \,|\, \varepsilon)$ can be factorized. See Yin and Zhou [2018] for a discussion on the differences between hierarchical variational models and SIVI.

$q_\theta(z \,|\, \varepsilon)$ to derive unbiased estimates of the gradient of the ELBO (see Section 2.2).

Example: Gaussian conditional. As a simple example, consider a multivariate Gaussian distribution for the conditional $q_\theta(z \,|\, \varepsilon)$. The parameters of the Gaussian are its mean $\mu_\theta(\varepsilon)$ and covariance $\Sigma_\theta(\varepsilon)$. Both parameters are given by neural networks with parameters θ and input ε.

The Gaussian meets the two assumptions outlined above. It is reparameterizable because it is in the location-scale family; the sampling process

$$u \sim q(u) = \mathcal{N}(u \,|\, 0, I),$$
$$z = h_\theta(u \,;\, \varepsilon) = \mu_\theta(\varepsilon) + \Sigma_\theta(\varepsilon)^{1/2} u$$

generates a sample $z \sim q_\theta(z \,|\, \varepsilon)$.

Furthermore, the Gaussian density and the gradient of the log-density can be evaluated. The latter is

$$\nabla_z \log q_\theta(z \,|\, \varepsilon) = -\Sigma_\theta(\varepsilon)^{-1} \left(z - \mu_\theta(\varepsilon)\right).$$

2.2 Unbiased Gradient Estimator

Here we derive the unbiased gradient estimators of the ELBO. First, UIVI uses the reparameterization $z = h_\theta(u \,;\, \varepsilon)$ (Eq. 4) to rewrite the expectation in Eq. 1 as an expectation with respect to $q(\varepsilon)$ and $q(u)$,

$$\mathcal{L}(\theta) = \mathbb{E}_{q(\varepsilon)q(u)} \left[\log p(x, z) - \log q_\theta(z) \Big|_{z=h_\theta(u \,;\, \varepsilon)} \right].$$

To obtain the gradient of the ELBO with respect to θ, the gradient operator can now be pushed inside the expectation, as in the standard reparameterization method [Kingma and Welling, 2014, Titsias and Lázaro-Gredilla, 2014, Rezende et al., 2014]. This gives two terms: one corresponding to the model and one corresponding to the entropy,

$$\nabla_\theta \mathcal{L}(\theta) = \mathbb{E}_{q(\varepsilon)q(u)} \left[g_\theta^{\text{mod}}(\varepsilon, u) + g_\theta^{\text{ent}}(\varepsilon, u) \right]. \quad (5)$$

The term corresponding to the model is

$$g_\theta^{\text{mod}}(\varepsilon, u) \triangleq \nabla_z \log p(x, z) \Big|_{z=h_\theta(u \,;\, \varepsilon)} \nabla_\theta h_\theta(u \,;\, \varepsilon); \quad (6)$$

similarly, the term corresponding to the entropy is

$$g_\theta^{\text{ent}}(\varepsilon, u) \triangleq -\nabla_z \log q_\theta(z) \Big|_{z=h_\theta(u \,;\, \varepsilon)} \nabla_\theta h_\theta(u \,;\, \varepsilon). \quad (7)$$

To obtain this decomposition, we have applied the identity that the expected value of the score function is zero, $\mathbb{E}_{q_\theta(z)} \left[\nabla_\theta \log q_\theta(z)\right] = 0$, which reduces the variance of the estimator [Roeder et al., 2017].

UIVI estimates the model component in Eq. 6 using samples from $q(\varepsilon)$ and $q(u)$. However, estimating the entropy component in Eq. 7 is harder because the term $\nabla_z \log q_\theta(z)$ cannot be evaluated—the variational distribution $q_\theta(z)$ is an implicit distribution.

UIVI addresses this issue rewriting Eq. 7 as an expectation, therefore enabling Monte Carlo estimates of the entropy component of the gradient. In particular, UIVI rewrites as an expectation the intractable log-density gradient in Eq. 7,

$$\nabla_z \log q_\theta(z) = \mathbb{E}_{q_\theta(\varepsilon \,|\, z)} \left[\nabla_z \log q_\theta(z \,|\, \varepsilon) \right]. \quad (8)$$

We prove Eq. 8 below. This equation shows that the problematic gradient $\nabla_z \log q_\theta(z)$ can be expressed in terms of an expression that can be evaluated—the gradient of the log-conditional $\nabla_z \log q_\theta(z \,|\, \varepsilon)$ can be evaluated by assumption (see Section 2.1). UIVI rewrites the entropy term in Eq. 7 using Eq. 8,

$$g_\theta^{\text{ent}}(\varepsilon, u) = - \mathbb{E}_{q_\theta(\varepsilon' \,|\, z)} \left[\nabla_z \log q_\theta(z \,|\, \varepsilon') \right] \Big|_{z=h_\theta(u \,;\, \varepsilon)} \times \nabla_\theta h_\theta(u \,;\, \varepsilon). \quad (9)$$

(We use the notation ε' to make it explicit that this variable is different from ε.)

The expectation in Eqs. 8 and 9 is taken with respect to the distribution $q_\theta(\varepsilon \,|\, z) \propto q_\theta(z \,|\, \varepsilon) q(\varepsilon)$. We call this distribution the *reverse conditional*. Although the conditional $q_\theta(z \,|\, \varepsilon)$ has a simple form (by assumption, it is a reparameterizable distribution for which we can evaluate the density and its gradient), the reverse conditional is complex because the conditional $q_\theta(z \,|\, \varepsilon)$ is parameterized by deep neural networks that take ε as input. We show in Section 2.3 how to efficiently draw samples from the reverse conditional to obtain an estimator of the entropy component in Eq. 9. (The main idea is to reuse a sample from the reverse conditional to initialize a sampler.)

We now prove Eq. 8 and then we show two examples that particularize these expressions for two choices of the conditionals: a multivariate Gaussian and a more general exponential family distribution.

Proof of Eq. 8. We show how to express the gradient $\nabla_z \log q_\theta(z)$ as an expectation. We start with the log-derivative identity,

$$\nabla_z \log q_\theta(z) = \frac{1}{q_\theta(z)} \nabla_z q_\theta(z).$$

Next we use the definition of the semi-implicit distribution $q_\theta(z)$ through a mixing distribution (Eq. 3) and we push the gradient into the integral,

$$\nabla_z \log q_\theta(z) = \frac{1}{q_\theta(z)} \nabla_z \int q_\theta(z \,|\, \varepsilon) q(\varepsilon) d\varepsilon$$
$$= \frac{1}{q_\theta(z)} \int \nabla_z q_\theta(z \,|\, \varepsilon) q(\varepsilon) d\varepsilon.$$

We now apply the log-derivative identity on the conditional $q_\theta(z \,|\, \varepsilon)$,

$$\nabla_z \log q_\theta(z) = \frac{1}{q_\theta(z)} \int q_\theta(z \,|\, \varepsilon) q(\varepsilon) \nabla_z \log q_\theta(z \,|\, \varepsilon) d\varepsilon.$$

Finally, we apply Bayes' theorem to obtain Eq. 8. □

Example: Gaussian conditional. Consider the multivariate Gaussian example from Section 2.1. Substituting the gradient of the Gaussian log-density into Eq. 9, we can write the entropy component of the gradient as

$$g_\theta^{\text{ent}}(\varepsilon, u) = \mathbb{E}_{q_\theta(\varepsilon' \mid z)}\left[\Sigma_\theta(\varepsilon')^{-1}\left(z - \mu_\theta(\varepsilon')\right)\right]\Big|_{z=h_\theta(u\,;\,\varepsilon)} \times \nabla_\theta h_\theta(u\,;\,\varepsilon).$$

Example: Exponential family conditional. Now consider the more general example of a reparameterizable exponential family conditional distribution $q_\theta(z \mid \varepsilon)$ with sufficient statistics $t(z)$ and natural parameter[2] $\eta_\theta(\varepsilon)$,

$$q_\theta(z \mid \varepsilon) \propto \exp\{t(z)^\top \eta_\theta(\varepsilon)\}. \tag{10}$$

Substituting the gradient $\nabla_z \log q_\theta(z \mid \varepsilon)$ into Eq. 9, we can obtain the entropy component of the gradient for a general (reparameterizable) exponential family distribution,

$$g_\theta^{\text{ent}}(\varepsilon, u) = -\nabla_z t(z)^\top \mathbb{E}_{q_\theta(\varepsilon' \mid z)}\left[\eta_\theta(\varepsilon')\right]\Big|_{z=h_\theta(u\,;\,\varepsilon)} \times \nabla_\theta h_\theta(u\,;\,\varepsilon).$$

2.3 Full Algorithm

UIVI estimates the gradient of the ELBO using Eq. 5, which decomposes the gradient as the expectation of the sum of the model component and the entropy component. UIVI estimates the expectation using S samples from $q(\varepsilon)$ and $q(u)$ ($S = 1$ in practice); that is,

$$\nabla_\theta \mathcal{L}(\theta) \approx \frac{1}{S}\sum_{s=1}^{S}\left(g_\theta^{\text{mod}}(\varepsilon_s, u_s) + g_\theta^{\text{ent}}(\varepsilon_s, u_s)\right), \quad \varepsilon_s \sim q(\varepsilon), \quad u_s \sim q(u).$$

The model component is given in Eq. 6 and the entropy component is given in Eq. 9. While the model component can be evaluated (the gradients involved can be obtained using autodifferentiation tools), the entropy component is more challenging because Eq. 9 contains an expectation with respect to the reverse conditional $q_\theta(\varepsilon \mid z)$. As this expectation is intractable, UIVI forms a Monte Carlo estimator using samples ε'_s from the reverse conditional.

The reverse conditional is a complex distribution due to the complex dependency of the (direct) conditional $q_\theta(z \mid \varepsilon)$ on the random variable ε. Consequently, sampling from the reverse conditional may be challenging.

UIVI exploits the fact that the samples ε_s that generated z_s are also samples from the reverse conditional. This is because the sampling procedure in Eq. 2 implies that each pair of samples (z_s, ε_s) comes from the joint $q_\theta(z, \varepsilon)$, and thus ε_s can be seen as a draw from the reverse conditional.

Algorithm 1 Unbiased implicit variational inference

Input: data x, semi-implicit variational family $q_\theta(z)$
Output: variational parameters θ
Initialize θ randomly
for iteration $t = 1, 2, \ldots,$ **do**
  `# Sample from q:`
  Sample $u_s \sim q(u)$ and $\varepsilon_s \sim q(\varepsilon)$
  Set $z_s = h_\theta(u_s\,;\,\varepsilon_s)$
  `# Sample from reverse conditional:`
  Sample $\varepsilon'_s \sim q_\theta(\varepsilon \mid z_s)$ (HMC initialized at ε_s)
  `# Estimate the gradient:`
  Compute $g_\theta^{\text{mod}}(\varepsilon_s, u_s)$ (Eq. 6)
  Compute $g_\theta^{\text{ent}}(\varepsilon_s, u_s)$ (Eq. 9, approximate using ε'_s)
  Compute $\widehat{\nabla}_\theta \mathcal{L} = g_\theta^{\text{mod}}(\varepsilon_s, u_s) + g_\theta^{\text{ent}}(\varepsilon_s, u_s)$
  `# Take gradient step:`
  Set $\theta \leftarrow \theta + \rho \cdot \widehat{\nabla}_\theta \mathcal{L}$
end for

Although ε_s is a valid sample from the reverse conditional $q_\theta(\varepsilon \mid z_s)$, setting $\varepsilon'_s = \varepsilon_s$ in the estimation of the entropy component (Eq. 9) would break the assumption that ε'_s and ε_s are independent. Instead, UIVI runs a Markov chain Monte Carlo (MCMC) method, such as Hamiltonian Monte Carlo (HMC) [Neal, 2011], to draw samples from the reverse conditional.[3] Crucially, UIVI initializes the MCMC chain at ε_s. In this way, there is no burn-in period in the MCMC procedure, in the sense that the sampler starts from stationarity so that any subsequent MCMC draw gives a sample from the reverse conditional [Robert and Casella, 2005]. To reduce the correlation between the sample ε'_s and the initialization value ε_s, UIVI runs more than one MCMC iterations and allows for a short burn-in period. (In the experiments of Section 4, we use 10 MCMC iterations where only the final 5 samples are used to form the Monte Carlo estimate.)

UIVI then forms an unbiased estimator of the entropy component (Eq. 9) using these samples from $q_\theta(\varepsilon \mid z_s)$,

$$g_\theta^{\text{ent}}(\varepsilon_s, u_s) \approx -\nabla_z \log q_\theta(z \mid \varepsilon'_s)\nabla_\theta h_\theta(u_s\,;\,\varepsilon_s), \quad \varepsilon'_s \sim q_\theta(\varepsilon \mid z_s), \qquad z_s = h_\theta(u_s\,;\,\varepsilon_s).$$

The full UIVI algorithm is summarized in Algorithm 1. For simplicity, in the description of the algorithm we assume one sample ε'_s for each sample ε_s; in practice we approximate each internal expectation under $q_\theta(\varepsilon' \mid z_s)$ with a few samples, i.e., the final 5 samples from each 10-length MCMC run as mentioned above. Code is publicly available.[4]

[2] We ignore the base measure in this definition; if needed it can be absorbed into $t(z)$.

[3] Note that this sampling algorithm does not require to evaluate the model $p(x, z)$, because the target distribution is $q_\theta(\varepsilon \mid z_s)$.

[4] See `https://github.com/franrruiz/uivi` for the code.

3 RELATED WORK

Among the methods to address the limitations of mean-field variational inference (VI), we can find methods that improve the mean-field posterior approximation using linear response estimates [Giordano et al., 2015, 2017], or methods that add dependencies among the latent variables using a structured variational family [Saul and Jordan, 1996], typically tailored to particular models [Ghahramani and Jordan, 1997, Titsias and Lázaro-Gredilla, 2011]. Other ways to add dependencies among the latent variables are mixtures [Bishop et al., 1998, Gershman et al., 2012, Salimans and Knowles, 2013, Guo et al., 2016, Miller et al., 2017], copulas [Tran et al., 2015, Han et al., 2016], hierarchical models [Ranganath et al., 2016, Tran et al., 2016, Maaløe et al., 2016], or general invertible transformations of random variables [Rezende et al., 2014, Kingma and Welling, 2014, Titsias and Lázaro-Gredilla, 2014, Kucukelbir et al., 2015, 2017], including normalizing flows [Rezende and Mohamed, 2015, Kingma et al., 2016, Papamakarios et al., 2017, Tomczak and Welling, 2016, 2017, Dinh et al., 2017]. Other approaches use spectral methods [Shi et al., 2018b] or define the variational distribution using sampling mechanisms [Salimans et al., 2015, Maddison et al., 2017, Naesseth et al., 2017, 2018, Le et al., 2018, Grover et al., 2018].

Implicit distributions develop a flexible variational family using non-invertible mappings parameterized by deep neural networks [Mohamed and Lakshminarayanan, 2016, Nowozin et al., 2016, Huszár, 2017, Tran et al., 2017, Li and Turner, 2018, Mescheder et al., 2017, Shi et al., 2018a]. The main issue of implicit distributions is density ratio estimation, which is often addressed using adversarial networks [Goodfellow et al., 2014]. However, density ratio estimation becomes particularly difficult in high-dimensional settings [Sugiyama et al., 2012].

The method that is more closely related to ours is semi-implicit variational inference (SIVI) [Yin and Zhou, 2018]. SIVI combines a simple reparameterizable distribution with an implicit one to obtain a flexible variational family. To find the variational parameters, SIVI maximizes a lower bound of the evidence lower bound (ELBO),

$$\mathcal{L}_{\text{SIVI}}^{(L)}(\theta) = \mathbb{E}_{\varepsilon\sim q(\varepsilon)}\Bigg[\mathbb{E}_{z\sim q_\theta(z\,|\,\varepsilon)}\Bigg[\mathbb{E}_{\varepsilon^{(1)},\ldots,\varepsilon^{(L)}\sim q(\varepsilon)}\Bigg[\log p(x,z) - \log\left(\frac{1}{L+1}\left(q_\theta(z\,|\,\varepsilon) + \sum_{\ell=1}^{L} q_\theta(z\,|\,\varepsilon^{(\ell)})\right)\right)\Bigg]\Bigg]\Bigg], \quad (11)$$

where $\mathcal{L}_{\text{SIVI}}^{(L)}(\theta) \leq \mathcal{L}(\theta)$ for any θ. At each iteration of the inference algorithm, the parameter L must form a non-decreasing sequence. As the parameter L grows to infinity, the lower bound $\mathcal{L}_{\text{SIVI}}^{(L)}$ approaches the ELBO in Eq. 1. The intuition behind the SIVI objective is to approximate the intractable marginalization $q_\theta(z) = \int q_\theta(z\,|\,\varepsilon)q(\varepsilon)d\varepsilon$, which appears in the entropy component of the ELBO, with $L+1$ draws from $q(\varepsilon)$.

Molchanov et al. [2019] have recently extended SIVI in the context of deep generative models. They use a semi-implicit construction of both the variational distribution and the deep generative model that defines the prior. This results in a doubly semi-implicit architecture that allows building a sandwich estimator of the ELBO. Similarly to SIVI, the objective of doubly semi-implicit variational inference is a bound on the ELBO that becomes tight as $L \to \infty$.

Finally, note that despite its similar name, the method of Figurnov et al. [2018] addresses a different problem. Specifically, it tackles the case where $q_\theta(z)$ is not reparameterizable but has a tractable density, for example a gamma or a beta distribution. In contrast, we address the problem where the variational distribution $q_\theta(z)$ is implicit.

4 EXPERIMENTS

We now apply unbiased implicit variational inference (UIVI) to assess the goodness of the resulting variational approximation and the computational complexity of the algorithm. As a baseline, we compare against semi-implicit variational inference (SIVI), which has been shown to outperform other approaches like mean-field variational inference (VI) and be on par with Markov chain Monte Carlo (MCMC) methods [Yin and Zhou, 2018].

First, in Section 4.1 we run toy experiments on simple two-dimensional distributions. Then, in Sections 4.2 and 4.3 we turn to more realistic models, including Bayesian multinomial logistic regression and the variational autoencoder (VAE) [Kingma and Welling, 2014].

4.1 Toy Experiments

To showcase UIVI, we approximate three synthetic distributions defined on a two-dimensional space: a banana distribution, a multimodal Gaussian, and an x-shaped Gaussian. Their densities are given in Table 1.

Variational family. To define the variational distribution, we choose a standard 3-dimensional Gaussian prior for $q(\varepsilon)$. We use a Gaussian conditional $q_\theta(z\,|\,\varepsilon) = \mathcal{N}(z\,|\,\mu_\theta(\varepsilon), \text{diag}(\sigma))$, whose mean is parameterized by a neural network with two hidden layers of 50 ReLu units each. We set a diagonal covariance that we also optimize (for simplicity, the covariance does not depend on ε). Thus, the variational parameters are the neural network weights and intercepts (θ) and the variances (σ).

Experimental setup. We run 50,000 iterations of Algorithm 1. We run 10 Hamiltonian Monte Carlo (HMC) iterations to draw samples from the reverse conditional $q_\theta(\varepsilon\,|\,z)$ (5 for burn-in and 5 actual samples), with 5 leapfrog steps [Neal, 2011]. We set the stepsize using RMSProp

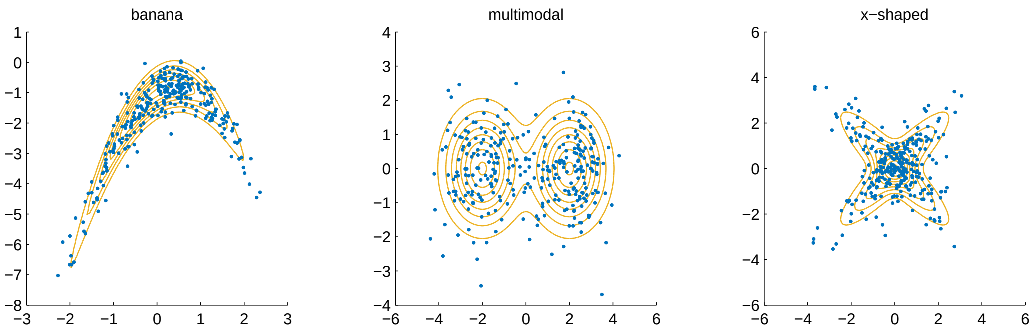


Figure 2. The samples from the variational distribution fitted with UIVI (blue) match the shape of the true synthetic target distributions (orange) considered in Section 4.1.

name	$p(z)$
banana	$\mathcal{N}\left(\left[\begin{array}{c} z_1 \\ z_2+z_1^2+1 \end{array}\right] \middle\| \left[\begin{array}{c} 0 \\ 0 \end{array}\right], \left[\begin{array}{cc} 1 & 0.9 \\ 0.9 & 1 \end{array}\right]\right)$
multimodal	$0.5\mathcal{N}\left(z \middle\| \left[\begin{array}{c} -2 \\ 0 \end{array}\right], I\right) + 0.5\mathcal{N}\left(z \middle\| \left[\begin{array}{c} 2 \\ 0 \end{array}\right], I\right)$
x-shaped	$0.5\mathcal{N}\left(z \,\middle\|\, 0, \left[\begin{array}{cc} 2 & 1.8 \\ 1.8 & 2 \end{array}\right]\right) + 0.5\mathcal{N}\left(z \,\middle\|\, 0, \left[\begin{array}{cc} 2 & -1.8 \\ -1.8 & 2 \end{array}\right]\right)$

Table 1. Synthetic distributions used in the toy experiment.

[Tieleman and Hinton, 2012]; at each iteration t we set $\rho^{(t)} = \eta/(1+\sqrt{G^{(t)}})$, where η is the learning rate, and the updates of $G^{(t)}$ depend on the gradient $\widehat{\nabla}_\theta \mathcal{L}^{(t)}$ as $G^{(t)} = 0.9G^{(t-1)} + 0.1(\widehat{\nabla}_\theta \mathcal{L}^{(t)})^2$. We set the learning rate $\eta = 0.01$ for the network parameters and $\eta = 0.002$ for the covariance, and we additionally decrease the learning rate by a factor of 0.9 every 3,000 iterations.

Results. Figure 2 shows the contour plot of the synthetic distributions, together with 300 samples from the fitted variational distribution. UIVI produces samples that match well the shape of the target distributions.

4.2 Bayesian Multinomial Logistic Regression

We now consider Bayesian multinomial logistic regression. For a dataset of N features x_n and labels $y_n \in \{1, \ldots, K\}$, the model is $p(z)\prod_n p(y_n \,|\, x_n, z)$, where z denotes the latent weights and biases. We set the prior $p(z)$ to be Gaussian with identity covariance and zero mean; the categorical likelihood is $p(y_n = k \,|\, x_n, z) \propto \exp(x_n^\top z_k + z_{0k})$.

Datasets. We use two datasets, MNIST[5] and HAPT,[6] both available online. MNIST contains 60,000 training and 10,000 test instances of 28×28 images of hand-written digits; thus there are $K = 10$ classes. We divide pixel values by 255 so that each feature is bounded between 0 and 1. HAPT [Reyes-Ortiz et al., 2016] is a human activity recognition dataset. It contains 7,767 training and 3,162 test 561-dimensional measurements captured by the sensors on a smartphone. There are $K = 12$ activities, including static postures (e.g., standing), dynamic activities (e.g., walking), and postural transitions (e.g., stand-to-sit).

Variational family. We use the variational family described in Section 4.1, namely, a Gaussian prior $q(\varepsilon)$ and Gaussian conditional $q_\theta(z \,|\, \varepsilon)$ with diagonal covariance. We set the dimensionality of ε to 100, and we use 200 hidden units on each of the two hidden layers of the neural network that parameterizes the mean of the Gaussian conditional.

Experimental setup. We run 100,000 iterations of UIVI, with the same experimental setup described in Section 4.1. To speed up the procedure, we subsample minibatches of data at each iteration of the algorithm [Hoffman et al., 2013]. We use a minibatch size of 2,000 for MNIST and 863 for HAPT. For the comparison with SIVI, we set the parameter $L = 200$ in Eq. 11. We use the same initialization of the variational parameters for both SIVI and UIVI.

Results. We found that the time per iteration was comparable for both methods. On average, SIVI took 0.14 seconds per iteration on MNIST and 0.09 seconds on HAPT, while UIVI took 0.11 and 0.10 seconds, respectively.

We obtain a Monte Carlo estimate of the evidence lower bound (ELBO) every 100 iterations (we use 100 samples, and we use 10,000 samples from $q(\varepsilon)$ to approximate the intractable entropy term). Figure 3 (top) shows the ELBO estimates; the plot has been smoothed using a rolling window of size 20 for easier visualization. UIVI provides a similar bound on the marginal likelihood than SIVI on MNIST and a slightly tighter bound on HAPT.

In addition, we also estimate the predictive log-likelihood on the test set every 1,000 iterations (we use 8,000 samples from the variational distribution to form this estimator). Figure 3 (bottom) shows the test log-likelihood as a function of the wall-clock time for both methods and datasets; the plot has been smoothed with a rolling window of size 2. UIVI achieves better predictions on both datasets.

Finally, we study the impact of the number of HMC iterations on the results. In Figure 4, we plot the ELBO as a

[5] http://yann.lecun.com/exdb/mnist

[6] https://archive.ics.uci.edu/ml/datasets/Smartphone-Based+Recognition+of+Human+Activities+and+Postural+Transitions

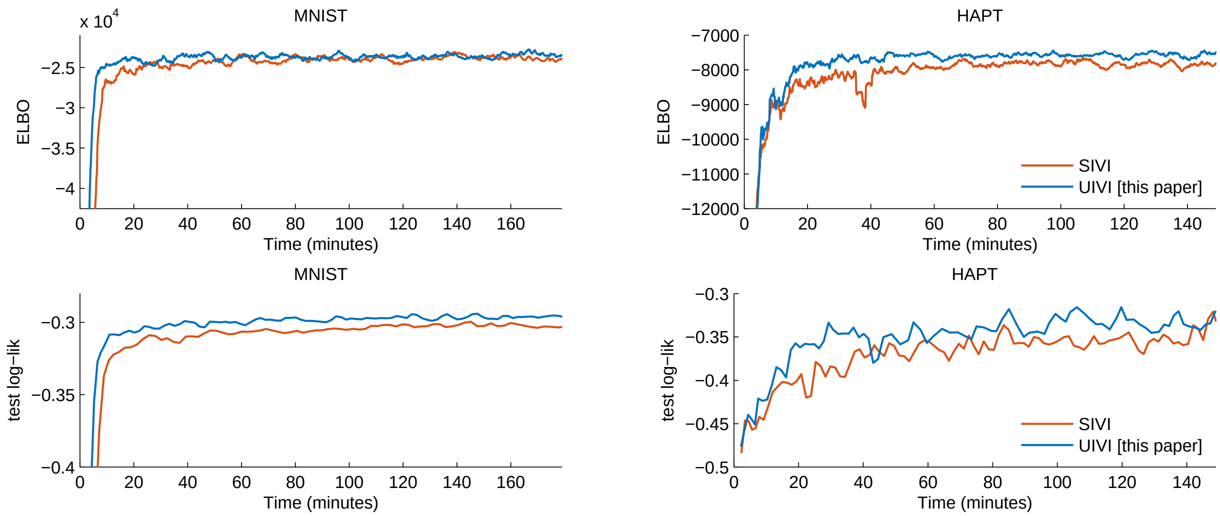


Figure 3. Estimates of the ELBO and the test log-likelihood as a function of wall-clock time for the Bayesian multinomial logistic regression model (Section 4.2). Compared to SIVI (red), UIVI (blue) achieves a better bound on the marginal likelihood and has better predictive performance.

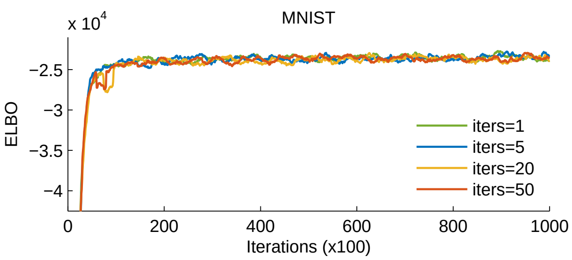


Figure 4. Estimates of the ELBO for the Bayesian multinomial logistic regression model (Section 4.2), obtained with UIVI under four different settings, which differ only in the number of HMC iterations. The number of HMC iterations in UIVI has a small impact on the results.

function of the iterations of the variational algorithm in four different settings on the MNIST dataset. Each of these settings corresponds to a different number of HMC iterations for both burn-in and sampling periods, ranging from 1 to 50 iterations (the standard setting of Figure 3 corresponds to 5 iterations). We conclude that the number of HMC iterations does not have a significant impact on the results. (Although not included here, the plot for the test log-likelihood shows no significant differences either.)

4.3 Variational Autoencoders

The VAE [Kingma and Welling, 2014] defines a conditional likelihood $p_\phi(x_n \,|\, z_n)$ given the latent variable z_n, parameterized by a neural network with parameters ϕ. The goal is to learn the parameters ϕ, for which the VAE introduces an amortized variational distribution $q_\theta(z_n \,|\, x_n)$. In the standard VAE, the variational distribution is Gaussian; we use instead a semi-implicit variational distribution.

Datasets. We use two datasets: (i) the binarized MNIST data [Salakhutdinov and Murray, 2008], which contains 50,000 training images and 10,000 test images of handwritten digits; and (ii) the binarized Fashion-MNIST data [Xiao et al., 2017], which contains 60,000 training images and 10,000 test images of clothing items. We binarize the Fashion-MNIST images with a threshold at 0.5. Images in both datasets are of size 28×28 pixels.

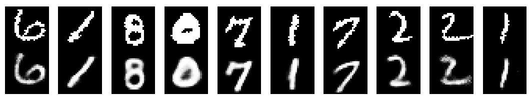
Variational family. We use the variational family described in Section 4.1 with Gaussian prior and Gaussian conditional. Since the variational distribution is amortized, we let the conditional $q_\theta(z_n \,|\, \varepsilon_n, x_n)$ depend on the observation x_n, such that the variational distribution is $q_\theta(z_n \,|\, x_n) = \int q(\varepsilon_n) q_\theta(z_n \,|\, \varepsilon_n, x_n) d\varepsilon_n$. We obtain the mean of the Gaussian conditional as the output of a neural network having as inputs both x_n and ε_n. We set the dimensionality of ε_n to 10 and the width of each the two hidden layers of the neural network to 200.

For comparisons, we also fit a standard VAE [Kingma and Welling, 2014]. The standard VAE uses an explicit Gaussian distribution whose mean and covariance are functions of the input, i.e., $q_\theta(z_n \,|\, x_n) = \mathcal{N}(z_n \,|\, \mu_\theta(x_n), \Sigma_\theta(x_n))$. The mean and covariance are parameterized using two separate neural networks with the same structure described above, and the covariance is set to be diagonal. The neural network for the covariance has softplus activations in the output layer, i.e., $\text{softplus}(x) = \log(1 + e^x)$.

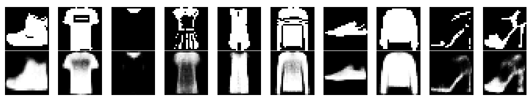
Experimental setup. For the generative model $p_\phi(x_n \,|\, z_n)$ we use a factorized Bernoulli distribution. We use a two-hidden-layer neural network with 200 hidden units on each hidden layer, whose sigmoidal outputs define the means of the Bernoulli distribution. We set the prior $p(z_n) = \mathcal{N}(z_n \,|\, 0, I)$ and the dimensionality of z_n to 10. We run

method	average test log-likelihood MNIST	Fashion-MNIST
Explicit (standard VAE)	-98.29	-126.73
SIVI	-97.77	-121.53
UIVI [this paper]	$\mathbf{-94.09}$	$\mathbf{-110.72}$

Table 2. Estimates of the marginal log-likelihood on the test set for the VAE (Section 4.3). UIVI gives better predictive performance than SIVI.



(a) MNIST images.



(b) Fashion-MNIST images.

Figure 5. Ten images reconstructed with the VAE model fitted with UIVI (Section 4.3). For each dataset, the top row shows training instances; the bottom row corresponds to the reconstructed images.

$400{,}000$ iterations of each method (explicit variational distribution, SIVI, and UIVI), using the same initialization and a minibatch of size 100. We set the SIVI parameter $L = 100$ so that both SIVI and UIVI have similar complexity (see below). We set the learning rate $\eta = 10^{-3}$ for the network parameters of the variational Gaussian conditional, $\eta = 2 \cdot 10^{-4}$ for its covariance (we also set $\eta = 2 \cdot 10^{-4}$ for the network that parameterizes the covariance of the explicit distribution), and $\eta = 10^{-3}$ for the network parameters of the generative model. We reduce the learning rate by a factor of 0.9 every $15{,}000$ iterations.

Results. We estimate the marginal likelihood on the test set using importance sampling,

$$\log p(x_n) \approx \log \frac{1}{S} \sum_{s=1}^{S} \frac{p_\phi(x_n \,|\, z_n^{(s)}) p(z_n^{(s)})}{\frac{1}{M} \sum_{m=1}^{M} q_\theta(z_n^{(s)} \,|\, \varepsilon_n^{(m)}, x_n)},$$
$$z_n^{(s)} \sim q_\theta(z_n \,|\, x_n), \quad \varepsilon_n^{(m)} \sim q(\varepsilon),$$

where we set $S = 1{,}000$ and $M = 10{,}000$ samples.

Table 2 shows the estimated values of the test marginal likelihood for all methods and datasets. UIVI provides better predictive performance than SIVI, which in turn gives better predictions than the explicit Gaussian approximation.

In terms of computational complexity, the average time per iteration is similar for UIVI and SIVI. On MNIST, it is 0.14 seconds for UIVI and 0.16 seconds for SIVI; on Fashion-MNIST, it is 0.13 seconds for UIVI and 0.17 for SIVI.

Finally, we show in Figure 5 ten training images from each dataset, together with the corresponding images reconstructed using the VAE fitted with UIVI. We reconstruct an image by first sampling $z_n \sim q_\theta(z_n \,|\, x_n)$ and then setting the reconstructed $\hat{x}_n$ to the mean given by the generative model $p_\phi(x_n \,|\, z_n)$. We conclude that UIVI is an effective method to optimize the VAE model.

5 CONCLUSION

We have developed unbiased implicit variational inference (UIVI), a method to approximate a target distribution with an expressive variational distribution. The variational distribution is implicit, and it is obtained through a reparameterizable distribution whose parameters follow a flexible distribution, similarly to semi-implicit variational inference (SIVI) [Yin and Zhou, 2018]. In contrast to SIVI, UIVI directly optimizes the evidence lower bound (ELBO) rather than a bound. For that, UIVI expresses the gradient of the ELBO as an expectation, enabling Monte Carlo estimates of the gradient. Compared to SIVI, we show that UIVI achieves better ELBO and predictive performance for Bayesian multinomial logistic regression and variational autoencoder.

Acknowledgements

Francisco J. R. Ruiz is supported by the EU Horizon 2020 programme (Marie Skłodowska-Curie Individual Fellowship, grant agreement 706760).

References

C. M. Bishop, N. D. Lawrence, T. S. Jaakkola, and M. I. Jordan. Approximating posterior distributions in belief networks using mixtures. In *Advances in Neural Information Processing Systems*, 1998.

D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

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

L. Dinh, J. Sohl-Dickstein, and S. Bengio. Density estimation using real NVP. In *International Conference on Learning Representations*, 2017.

M. Figurnov, S. Mohamed, and A. Mnih. Implicit reparameterization gradients. In *Advances in Neural Information Processing Systems*, 2018.

S. J. Gershman, M. D. Hoffman, and D. M. Blei. Nonparametric variational inference. In *International Conference on Machine Learning*, 2012.

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

Z. Ghahramani and M. I. Jordan. Factorial hidden Markov models. *Machine Learning*, 29(2–3):245–273, 1997.

R. J. Giordano, T. Broderick, and M. I. Jordan. Linear response methods for accurate covariance estimates from mean field variational Bayes. In *Advances in Neural Information Processing Systems*, 2015.

R. J. Giordano, T. Broderick, and M. I. Jordan. Covariances, robustness, and variational Bayes. In *arXiv:1709.02536*, 2017.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.

A. Grover, R. Gummadi, M. Lázaro-Gredilla, D. Schuurmans, and S. Ermon. Variational rejection sampling. In *Artificial Intelligence and Statistics*, 2018.

F. Guo, X. Wang, K. Fan, T. Broderick, and D. B. Dunson. Boosting variational inference. In *arXiv:1611.05559*, 2016.

S. Han, X. Liao, D. B. Dunson, and L. Carin. Variational Gaussian copula inference. In *Artificial Intelligence and Statistics*, 2016.

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

F. Huszár. Variational inference using implicit distributions. In *arXiv:1702.08235*, 2017.

T. S. Jaakkola and M. I. Jordan. Improving the mean field approximation via the use of mixture distributions. In *Learning in Graphical Models*, pages 163–173, 1998.

M. I. Jordan, editor. *Learning in Graphical Models*. MIT Press, Cambridge, MA, USA, 1999.

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

D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, and M. Welling. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems*, 2016.

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. *Journal of Machine Learning Research*, 18(14):1–45, 2017.

T. A. Le, M. Igl, T. Rainforth, T. Jin, and F. Wood. Auto-encoding sequential Monte-Carlo. In *International Conference on Learning Representations*, 2018.

Y. Li and R. E. Turner. Gradient estimators for implicit models. In *International Conference on Learning Representations*, 2018.

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

C. J. Maddison, D. Lawson, G. Tucker, N. Heess, M. Norouzi, A. Mnih, A. Doucet, and Y. W. Teh. Filtering variational objectives. In *Advances in Neural Information Processing Systems*, 2017.

L. Mescheder, S. Nowozin, and A. Geiger. Adversarial variational Bayes: Unifying variational autoencoders and generative adversarial networks. In *International Conference on Machine Learning*, 2017.

A. C. Miller, N. Foti, and R. P. Adams. Variational boosting: Iteratively refining posterior approximations. In *International Conference on Machine Learning*, 2017.

S. Mohamed and B. Lakshminarayanan. Learning in implicit generative models. In *arXiv:1610.03483*, 2016.

D. Molchanov, V. Kharitonov, A. Sobolev, and D. Vetrov. Doubly semi-implicit variational inference. In *Artificial Intelligence and Statistics*, 2019.

C. Naesseth, F. J. R. Ruiz, S. Linderman, and D. M. Blei. Reparameterization gradients through acceptance-rejection methods. In *Artificial Intelligence and Statistics*, 2017.

C. Naesseth, S. W. Linderman, R. Ranganath, and D. M. Blei. Variational sequential Monte Carlo. In *Artificial Intelligence and Statistics*, 2018.

R. M. Neal. MCMC using Hamiltonian dynamics. In S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng, editors, *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC, 2011.

S. Nowozin, B. Cseke, and R. Tomioka. f-GAN: training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, 2016.

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

G. Papamakarios, I. Murray, and T. Pavlakou. Masked autoregressive flow for density estimation. In *Advances in Neural Information Processing Systems*, 2017.

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

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

J. L. Reyes-Ortiz, L. Oneto, A. Samà, X. Parra, and D. Anguita. Transition-aware human activity recognition using smartphones. *Neurocomputing*, 171(C):754–767, jan 2016.

D. J. Rezende and S. Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, 2015.

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

G. Roeder, Y. Wu, and D. Duvenaud. Sticking the landing: Simple, lower-variance gradient estimators for variational inference. In *Advances in Neural Information Processing Systems*, 2017.

R. Salakhutdinov and I. Murray. On the quantitative analysis of deep belief networks. In *International Conference on Machine Learning*, 2008.

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.

L. K. Saul and M. I. Jordan. Exploiting tractable substructures in intractable networks. In *Advances in Neural Information Processing Systems*, 1996.

J. Shi, S. Sun, and J. Zhu. Kernel implicit variational inference. In *International Conference on Learning Representations*, 2018a.

J. Shi, S. Sun, and J. Zhu. A spectral approach to gradient estimation for implicit distributions. In *International Conference on Machine Learning*, 2018b.

M. Sugiyama, T. Suzuki, and T. Kanamori. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

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. Spike and slab variational inference for multi-task and multiple kernel learning. In *Advances in Neural Information Processing Systems*, 2011.

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

J. M. Tomczak and M. Welling. Improving variational auto-encoders using convex combination linear inverse autoregressive flow. In *arXiv:1706.02326*, 2016.

J. M. Tomczak and M. Welling. Improving variational auto-encoders using Householder flow. In *arXiv:1611.09630*, 2017.

D. Tran, D. M. Blei, and E. M. Airoldi. Copula variational inference. In *Advances in Neural Information Processing Systems*, 2015.

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

D. Tran, R. Ranganath, and D. M. Blei. Hierarchical implicit models and likelihood-free variational inference. In *Advances in Neural Information Processing Systems*, 2017.

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

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. In *arXiv:1708.07747*, 2017.

M. Yin and M. Zhou. Semi-implicit variational inference. In *International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5660–5669. PMLR, 2018.

C. Zhang, J. Bütepage, H. Kjellström, and S. Mandt. Advances in variational inference. *arXiv:1711.05597*, 2017.