
Hamiltonian Variational Auto-Encoder

Anthony L. Caterini, Arnaud Doucet, Dino Sejdinovic

Department of Statistics

University of Oxford

{anthony.caterini, doucet, dino.sejdinovic}@stats.ox.ac.uk

Abstract

Variational Auto-Encoders (VAEs) have become very popular techniques to perform inference and learning in latent variable models as they allow us to leverage the rich representational power of neural networks to obtain flexible approximations of the posterior of latent variables as well as tight evidence lower bounds (ELBOs). Combined with stochastic variational inference, this provides a methodology scaling to large datasets. However, for this methodology to be practically efficient, it is necessary to obtain low-variance unbiased estimators of the ELBO and its gradients with respect to the parameters of interest. While the use of Markov chain Monte Carlo (MCMC) techniques such as Hamiltonian Monte Carlo (HMC) has been previously suggested to achieve this [23, 26], the proposed methods require specifying reverse kernels which have a large impact on performance. Additionally, the resulting unbiased estimator of the ELBO for most MCMC kernels is typically not amenable to the reparameterization trick. We show here how to optimally select reverse kernels in this setting and, by building upon Hamiltonian Importance Sampling (HIS) [17], we obtain a scheme that provides low-variance unbiased estimators of the ELBO and its gradients using the reparameterization trick. This allows us to develop a Hamiltonian Variational Auto-Encoder (HVAE). This method can be re-interpreted as a target-informed normalizing flow [20] which, within our context, only requires a few evaluations of the gradient of the sampled likelihood and trivial Jacobian calculations at each iteration.

1 Introduction

Variational Auto-Encoders (VAEs), introduced in Kingma and Welling [13] and Rezende et al. [21], are popular techniques to carry out inference and learning in complex latent variable models. However, the standard mean-field parametrization of the approximate posterior distribution can limit its flexibility. Recent work has sought to *augment* the VAE approach by sampling from the VAE posterior approximation and transforming these samples through mappings with additional trainable parameters to achieve richer posterior approximations. The most popular application of this idea is the Normalizing Flows (NFs) approach [20], in which the samples are deterministically evolved through a series of parameterized invertible transformations called a *flow*. NFs have demonstrated success in various domains [2, 14], but their form of the flow does not explicitly use information about the target posterior, so it is unclear whether an improved performance is due to an accurate posterior approximation or simply a result of overparametrization. The related Hamiltonian Variational Inference (HVI) [23] instead stochastically evolves the base samples according to Hamiltonian Monte Carlo (HMC) [18] and thus uses target information, but relies on defining reverse dynamics in the flow, which, as we will see, turns out to be unnecessary and suboptimal.

One of the key components in the formulation of VAEs is the maximization of the evidence lower bound (ELBO). The main idea put forward in Salimans et al. [23] is that it is possible to use K MCMC iterations to obtain an unbiased estimator of the ELBO and its gradients. This estimator

is obtained using an importance sampling argument on an augmented space, with the importance distribution being the joint distribution of the $K + 1$ states of the ‘forward’ Markov chain, while the augmented target distribution is constructed using a sequence of ‘reverse’ Markov kernels such that it admits the original posterior distribution as a marginal. The performance of this estimator is strongly dependent on the selection of these forward and reverse kernels, but no clear guideline for selection has been provided therein. By linking this approach to earlier work by Del Moral et al. [6], we show how to select these components. We focus, in particular, on the use of time-inhomogeneous Hamiltonian dynamics, proposed originally in Neal [17], to augment underlying VAE models. This method uses reverse kernels which are optimal for reducing variance of the log-likelihood estimators and allows for simple calculation of the approximate log-posteriors after evolution through the dynamics. Additionally, we can easily use the reparameterization trick to calculate unbiased gradients of the ELBO with respect to parameters of interest. The resulting method, which we refer to as the Hamiltonian Variational Auto-Encoder (HVAE), can be thought of as a Normalizing Flow scheme in which the flow depends explicitly on the target distribution. This combines the best properties of HVI and NFs, resulting in target-informed and inhomogeneous deterministic Hamiltonian dynamics, while being scalable to large datasets and high dimensions.

2 Evidence Lower Bounds, MCMC and Hamiltonian Importance Sampling

2.1 Unbiased likelihood and evidence lower bound estimators

For data $x \in \mathcal{X} \subseteq \mathbb{R}^d$ and parameter $\theta \in \Theta$, consider the likelihood function

$$p_\theta(x) = \int p_\theta(x, z) dz = \int p_\theta(x|z)p_\theta(z) dz,$$

where $z \in \mathcal{Z}$ are some latent variables. Assume we have access to a strictly positive unbiased estimate $\hat{p}_\theta(x)$ of $p_\theta(x)$, then

$$\int \hat{p}_\theta(x) q_{\theta, \phi}(u|x) du = p_\theta(x), \quad (1)$$

with $u \sim q_{\theta, \phi}(\cdot)$, $u \in \mathcal{U}$ denoting all the random variables used to compute $\hat{p}_\theta(x)$. Here, ϕ denotes additional parameters of the sampling distribution. We emphasize that $\hat{p}_\theta(x)$ depends on both u and potentially ϕ as this is not done notationally. By applying Jensen’s inequality to (1), we thus obtain, for all θ and ϕ ,

$$\mathcal{L}_{\text{ELBO}}(\theta, \phi; x) := \int \log \hat{p}_\theta(x) q_{\theta, \phi}(u|x) du \leq \log p_\theta(x) := \mathcal{L}(\theta; x). \quad (2)$$

It can be shown that $|\mathcal{L}_{\text{ELBO}}(\theta, \phi; x) - \mathcal{L}(\theta; x)|$ decreases as the variance of $\hat{p}_\theta(x)$ decreases; see, e.g., [3, 15]. The standard variational framework corresponds to $\mathcal{U} = \mathcal{Z}$ and $\hat{p}_\theta(x) = p_\theta(x, z)/q_{\theta, \phi}(z|x)$, while the Importance Weighted Auto-Encoder (IWAE) [3] corresponds to $\mathcal{U} = \mathcal{Z}^L$, $q_{\theta, \phi}(u|x) = \prod_{i=1}^L q_{\theta, \phi}(z_i|x)$ and $\hat{p}_\theta(x) = \frac{1}{L} \sum_{i=1}^L p_\theta(x, z_i)/q_{\theta, \phi}(z_i|x)$.

In the general case, we do not have an analytical expression for $\mathcal{L}_{\text{ELBO}}(\theta, \phi; x)$, but when performing stochastic gradient variational inference we only require an unbiased estimator of $\nabla_\theta \mathcal{L}_{\text{ELBO}}(\theta, \phi; x)$. This is given by $\nabla_\theta \log \hat{p}_\theta(x)$ if the reparameterization trick [8, 13] is used, i.e. $q_{\theta, \phi}(u|x) = q(u)$, and $\hat{p}_\theta(x)$ is a smooth enough function of u . As a guiding principle, one should attempt to obtain a low-variance estimator of $\hat{p}_\theta(x)$, which typically translates into a low-variance estimator of $\nabla_\theta \mathcal{L}_{\text{ELBO}}(\theta, \phi; x)$. We can analogously optimize $\mathcal{L}_{\text{ELBO}}(\theta, \phi; x)$ with respect to ϕ through stochastic gradient ascent to obtain tighter bounds.

2.2 Unbiased likelihood estimator using MCMC

Salimans et al. [23] propose to build an unbiased estimator of $\hat{p}_\theta(x)$ by sampling a (potentially time-inhomogeneous) ‘forward’ Markov chain of length $K + 1$ using $z_0 \sim q_{\theta, \phi}^0(\cdot)$ and $z_k \sim q_{\theta, \phi}^k(\cdot|z_{k-1})$ for $k = 1, \dots, K$. Using artificial ‘reverse’ Markov transition kernels $r_{\theta, \phi}^k(z_k|z_{k+1})$ for $k = 0, \dots, K - 1$, it follows easily from an importance sampling argument that

$$\hat{p}_\theta(x) = \frac{p_\theta(x, z_K) \prod_{k=0}^{K-1} r_{\theta, \phi}^k(z_k|z_{k+1})}{q_{\theta, \phi}^0(z_0) \prod_{k=1}^K q_{\theta, \phi}^k(z_k|z_{k-1})} \quad (3)$$

is an unbiased estimator of $\hat{p}_\theta(x)$ as long as the ratio in (3) is well-defined. In the framework of the previous section, we have $\mathcal{U} = \mathcal{Z}^{K+1}$ and $q_{\theta,\phi}(u|x)$ is given by the denominator of (3). Although we did not use measure-theoretic notations, the kernels $q_{\theta,\phi}^k$ are typically MCMC kernels which do not admit a density with respect to the Lebesgue measure (e.g. the Metropolis–Hastings kernel). This makes it difficult to define reverse kernels for which (3) is well-defined as evidenced in Salimans et al. [23, Section 4] or Wolf et al. [26]. Del Moral et al. [6] – wherein the estimator (3) was originally introduced – provides generic recommendations for obtaining low-variance estimators of $p_\theta(x)$: select $q_{\theta,\phi}^k$ as MCMC kernels which are invariant, or approximately invariant, with respect to $p_\theta^k(x, z_k)$, where $p_{\theta,\phi}^k(z|x) \propto [q_{\theta,\phi}^0(z_0)]^{1-\beta_k} [p_\theta(x, z)]^{\beta_k}$ is a sequence of artificial densities bridging $q_{\theta,\phi}^0(z)$ to $p_\theta(z|x)$ smoothly using $\beta_0 = 0 < \beta_1 < \dots < \beta_{K-1} < \beta_K = 1$. It is also established in Del Moral et al. [6] that, given any sequence of kernels $\{q_{\theta,\phi}^k\}_k$, the sequence of reverse kernels minimizing the variance of $\hat{p}_\theta(x)$ is given by $r_{\theta,\phi}^{k,\text{opt}}(z_k|z_{k+1}) = q_{\theta,\phi}^k(z_k)q_{\theta,\phi}^{k+1}(z_{k+1}|z_k)/q_{\theta,\phi}^{k+1}(z_{k+1})$, where $q_{\theta,\phi}^k(z_k)$ denotes the marginal density of z_k under the forward dynamics, yielding

$$\hat{p}_\theta(x) = \frac{p_\theta(x, z_K)}{q_{\theta,\phi}^K(z_K)}. \quad (4)$$

For stochastic forward transitions, it is typically not possible to compute $r_{\theta,\phi}^{k,\text{opt}}$ and the corresponding estimator (4) as the marginal densities $q_{\theta,\phi}^k(z_k)$ do not admit closed-form expressions. However this suggests that $r_{\theta,\phi}^k$ should be approximating $r_{\theta,\phi}^{k,\text{opt}}$ and various schemes are presented in [6]. As noticed by Del Moral et al. [6] and Salimans et al. [23], Annealed Importance Sampling (AIS) [16] – also known as the Jarzynski–Crooks identity ([4, 11]) in physics – is a special case of (3) using, for $q_{\theta,\phi}^k$, a $p_\theta^k(z|x)$ -invariant MCMC kernel and tractable, but suboptimal, reverse transition kernels. AIS provides state-of-the-art estimators of the marginal likelihood and has been widely used in machine learning. Unfortunately, it typically cannot be used in conjunction with the reparameterization trick. Indeed, although it is very often possible to reparameterize the forward simulation of (z_1, \dots, z_T) in terms of the deterministic transformation of some random variables $u \sim q$ independent of θ , this mapping is not continuous because the MCMC kernels it uses typically include singular components. In this context, although (1) holds, $\nabla_\theta \log \hat{p}_\theta(x)$ is *not an unbiased estimator of $\nabla_\theta \mathcal{L}_{\text{ELBO}}(\theta; x)$* ; see, e.g., [8] for a careful discussion of these issues.

2.3 Using Hamiltonian dynamics

Given the empirical success of Hamiltonian Monte Carlo (HMC) [10, 18], various contributions have proposed to develop algorithms exploiting Hamiltonian dynamics to obtain unbiased estimates of the ELBO and its gradients when $\mathcal{Z} = \mathbb{R}^\ell$. This is indeed the basis of Salimans et al. [23]. However, the algorithm proposed therein relies on a time-homogeneous leapfrog where momentum resampling is performed at each step and no Metropolis correction is used. It also relies on a rather arbitrary reverse kernel. To address the limitations of Salimans et al. [23], Wolf et al. [26] have proposed to include some Metropolis acceptance steps, but they still limit themselves to homogeneous dynamics and their estimator is not amenable to the reparameterization trick. Finally, in Hoffman [9], an alternative approach is used where the gradient $\nabla \mathcal{L}(\theta; x)$ of the true likelihood is directly approximated by using Fisher’s identity and HMC to obtain approximate samples from $p_\theta(z|x)$. However, the MCMC bias can be very significant when one has multimodal latent posteriors, and is strongly dependent on both the initial distribution and θ .

Here, we follow an alternative approach where we use Hamiltonian dynamics that are time-inhomogeneous as in [6] and [16], and use optimal reverse Markov kernels to compute $\hat{p}_\theta(x)$. This estimator can be used in conjunction with the reparameterization trick to obtain an unbiased estimator of $\nabla \mathcal{L}_{\text{ELBO}}(\theta, \phi; x)$. This method is based on the Hamiltonian Importance Sampling (HIS) scheme proposed in Neal [17]; one can also find several instances of related ideas in physics [12, 24]. We work in an extended space $(z, \rho) \in \mathcal{U} := \mathbb{R}^\ell \times \mathbb{R}^\ell$, introducing *momentum* variables ρ to pair with the *position* variables z , with $\bar{p}_\theta(x, z, \rho) := p_\theta(x, z)\mathcal{N}(\rho|0, M)$ for some symmetric positive definite mass matrix $M \in \mathbb{R}^{\ell \times \ell}$. Essentially, the idea is to sample using *deterministic* transitions $q_{\theta,\phi}^k((z_k, \rho_k)|(z_{k-1}, \rho_{k-1})) = \delta_{\Phi_{\theta,\phi}^k(z_{k-1}, \rho_{k-1})}(z_k, \rho_k)$ so that $(z_K, \rho_K) = \mathcal{H}_{\theta,\phi}(z_0, \rho_0) := \Phi_{\theta,\phi}^K \circ \dots \circ \Phi_{\theta,\phi}^1(z_0, \rho_0)$, where $(z_0, \rho_0) \sim q_{\theta,\phi}^0(\cdot, \cdot)$ and $(\Phi_{\theta,\phi}^k)_{k \geq 1}$,

define diffeomorphisms corresponding to a time-discretized and inhomogeneous Hamiltonian dynamics. In this case, it is easy to show that

$$q_{\theta,\phi}^K(z_K, \rho_K) = q_{\theta,\phi}^0(z_0, \rho_0) \prod_{k=1}^K \det |\nabla \Phi_{\theta,\phi}^k(z_k, \rho_k)|^{-1} \quad \text{and} \quad \hat{p}_\theta(x) = \frac{\bar{p}_\theta(x, z_K, \rho_K)}{q_{\theta,\phi}^K(z_K, \rho_K)}. \quad (5)$$

It can also be shown that this is nothing but a special case of (3) (on the extended position-momentum space) using the optimal reverse kernels $r_{\theta,\phi}^{k,\text{opt}}$. This setup is similar to the one of normalizing flows [20], except here we use a flow informed by the target distribution. Indeed, Salimans et al. [23] is mentioned in Rezende and Mohamed [20], but the flow therein is homogeneous and yields a high-variance estimator of the normalizing constants even if $r_{\theta,\phi}^{k,\text{opt}}$ is used, as demonstrated in our simulations in section 4.

Note that we can write the estimator $\hat{p}_\theta(x)$ in (5) as

$$\hat{p}_\theta(x) = \frac{\bar{p}_\theta(x, \mathcal{H}_{\theta,\phi}(z_0, \rho_0))}{q_{\theta,\phi}^0(z_0, \rho_0)} \prod_{k=1}^K \det |\nabla \Phi_{\theta,\phi}^k(z_k, \rho_k)|. \quad (6)$$

Hence, if we can simulate $(z_0, \rho_0) \sim q_{\theta,\phi}^0(\cdot, \cdot)$ using $(z_0, \rho_0) = \Psi_{\theta,\phi}(u)$, where $u \sim q$ and $\Psi_{\theta,\phi}$ is a smooth mapping, then we can use the reparameterization trick since $\Phi_{\theta,\phi}^k$ are also smooth mappings.

In our case, the deterministic transformation $\Phi_{\theta,\phi}^k$ has two components: a leapfrog step, which discretizes the Hamiltonian dynamics, and a tempering step, which adds inhomogeneity to the dynamics and allows us to explore isolated modes of the target. To describe the leapfrog step, we first define the potential energy of the system as $U_\theta(z|x) \equiv -\log p_\theta(x, z)$ for a single datapoint $x \in \mathcal{X}$. Leapfrog then takes the system from (z, ρ) into (z', ρ') via the following transformations:

$$\tilde{\rho} = \rho - \frac{\varepsilon}{2} \odot \nabla U_\theta(z|x), \quad (7)$$

$$z' = z + \varepsilon \odot (M^{-1} \tilde{\rho}), \quad (8)$$

$$\rho' = \tilde{\rho} - \frac{\varepsilon}{2} \odot \nabla U_\theta(z'|x), \quad (9)$$

where $\varepsilon \in \mathbb{R}^\ell$ are the individual leapfrog step sizes per dimension, \odot denotes elementwise multiplication, and the gradient of $U_\theta(z|x)$ is taken with respect to z . The composition of equations (7) - (9) has unit Jacobian since each equation describes a shear transformation. As for the tempering part, we prescribe an inverse temperature schedule that increases from $\beta_0 \in (0, 1)$ initially to $\beta_K = 1$ at the end of each trajectory. We implement tempering at each step $k \in [K]^1$ by multiplying the momentum output of leapfrog by $\alpha_k \equiv \sqrt{\beta_{k-1}/\beta_k}$ before going to the next leapfrog step. This cooling operation has Jacobian α_k^ℓ . We obtain $\Phi_{\theta,\phi}^k$ by composing the leapfrog integrator with the cooling operation, which implies that the Jacobian is given by $\det |\nabla \Phi_{\theta,\phi}^k(z_k, \rho_k)| = \alpha_k^\ell = (\beta_{k-1}/\beta_k)^{\ell/2}$, which in turns implies

$$\prod_{k=1}^K \det |\nabla \Phi_{\theta,\phi}^k(z_k, \rho_k)| = \prod_{k=1}^K \left(\frac{\beta_{k-1}}{\beta_k} \right)^{\ell/2} = \beta_0^{\ell/2}.$$

The only remaining component to specify is the initial distribution. We will set $q_{\theta,\phi}^0(z_0, \rho_0) = q_{\phi_0}^0(z_0) \cdot \mathcal{N}(\rho_0|0, M/\beta_0)$, where $q_{\phi_0}^0(z_0)$ will be referred to as the *variational prior* with parameters ϕ_0 , and $\mathcal{N}(\rho_0|0, M/\beta_0)$ is the canonical momentum distribution at inverse temperature β_0 . The full procedure to generate an unbiased estimate of the ELBO from (2) on the extended space \mathcal{U} for a single point $x \in \mathcal{X}$ is given in Algorithm 1. The set of variational parameters to optimize is given by $\phi \equiv \phi_0 \cup \{\beta_0, \varepsilon, M\}$, although we will often avoid optimizing M as we can capture the same effect by optimizing individual leapfrog step sizes per dimension [18, Section 4.2]. We can see from (6) that we will obtain unbiased gradients with respect to θ and ϕ from our estimate of the ELBO if we write $(z_0, \rho_0) = (z_0, (M/\beta_0)^{1/2} \gamma_0)$, for $z_0 \sim q_{\phi_0}^0(\cdot)$ and $\gamma_0 \sim \mathcal{N}(\cdot|0, I_\ell) \equiv \mathcal{N}_\ell(\cdot)$, provided we are not also optimizing with respect to ϕ_0 . We will require additional reparameterization when we do elect to optimize with respect to ϕ_0 , but this is generally quite easy to implement on a problem-specific basis and is well-known in the literature.

¹ $[K] \equiv \{1, \dots, K\}$

Algorithm 1 Hamiltonian ELBO

Require: $p_\theta(x, \cdot)$ is the unnormalized posterior for $x \in \mathcal{X}$ and $\theta \in \Theta$

Require: $q_{\phi_0}^0(\cdot)$ is the variational prior on \mathbb{R}^ℓ

```

function HIS( $x, \theta, K, \beta_0, \varepsilon, M$ )
  Sample  $z_0 \sim q_{\phi_0}^0(\cdot), \gamma_0 \sim \mathcal{N}_\ell(\cdot)$ 
   $\rho_0 \leftarrow (M/\beta_0)^{1/2} \gamma_0$ 
  for  $k \leftarrow 1$  to  $K$  do
     $\tilde{\rho} \leftarrow \rho - \varepsilon/2 \odot \nabla U_\theta(z_{k-1}|x)$ 
     $z_k \leftarrow z_{k-1} + \varepsilon \odot (M^{-1} \tilde{\rho})$ 
     $\rho' \leftarrow \tilde{\rho} - \varepsilon/2 \odot \nabla U_\theta(z_k|x)$ 
     $\sqrt{\beta_k} \leftarrow \left( \left(1 - \frac{1}{\sqrt{\beta_0}}\right) \cdot k^2/K^2 + \frac{1}{\sqrt{\beta_0}} \right)^{-1}$ 
     $\rho_k \leftarrow \sqrt{\beta_{k-1}/\beta_k} \cdot \rho'$ 
   $\bar{p} \leftarrow p_\theta(x, z_K) \mathcal{N}(\rho_K|0, M)$ 
   $\bar{q} \leftarrow q_{\phi_0}^0(z_0) \mathcal{N}(\rho_0|0, M/\beta_0) \beta_0^{-\ell/2}$ 
   $\hat{\mathcal{L}}_{\text{ELBO}}^H(\theta, \phi; x) \leftarrow \log \bar{p} - \log \bar{q}$ 
return  $\hat{\mathcal{L}}_{\text{ELBO}}^H(\theta, \phi; x)$ 

```

\triangleright Note that $\rho_0 \sim \mathcal{N}(\cdot|0, M/\beta_0)$, reparameterized
 \triangleright Run K steps of alternating leapfrog and tempering
 \triangleright Start of leapfrog; equation (7)
 \triangleright Equation (8)
 \triangleright Equation (9)
 \triangleright Quadratic tempering scheme
 \triangleright Equation (5), left side
 \triangleright Take the log of equation (5), right side
 \triangleright Can take unbiased gradients of this estimate wrt θ, ϕ

3 Stochastic Variational Inference

We will now describe how to use Algorithm 1 within a stochastic variational inference procedure, moving to the setting where we have a dataset $\mathcal{D} = \{x_1, \dots, x_N\}$ and $x_i \in \mathcal{X}$ for all $i \in [N]$. In this case, we are interested in finding

$$\theta^* \in \operatorname{argmax}_{\theta \in \Theta} \mathbf{E}_{X \sim \delta_{\mathcal{D}}(\cdot)}[\mathcal{L}(\theta; X)], \quad (10)$$

where $\delta_{\mathcal{D}}(\cdot) \equiv \sum_{i=1}^N \delta_{x_i}(\cdot)/N$ is the empirical measure over the data. We resort to variational methods since $\mathcal{L}(\theta; X)$ cannot generally be calculated exactly and maximize instead the surrogate ELBO objective function

$$\mathcal{L}_{\text{ELBO}}(\theta, \phi) \equiv \mathbf{E}_{X \sim \delta_{\mathcal{D}}(\cdot)}[\mathcal{L}_{\text{ELBO}}(\theta, \phi; X)] \quad (11)$$

for $\mathcal{L}_{\text{ELBO}}(\theta, \phi; X)$ defined as in (2). We can now turn to stochastic gradient ascent (or a variant thereof) to jointly maximize (11) with respect to θ and ϕ by approximating the expectation over $\delta_{\mathcal{D}}(\cdot)$ using *minibatches* of observed data.

For our specific problem, we can reduce the variance of the ELBO calculation by analytically evaluating some terms in the expectation as follows:

$$\begin{aligned}
\mathcal{L}_{\text{ELBO}}^H(\theta, \phi) &= \mathbf{E}_{X \sim \delta_{\mathcal{D}}(\cdot)} \left[\mathbf{E}_{(Z_0, \rho_0) \sim q_{\phi_0}^0(\cdot, \cdot)} \left[\log \left(\frac{\bar{p}_\theta(X, Z_K, \rho_K) \beta_0^{\ell/2}}{q_\phi^0(Z_0, \rho_0)} \right) \right] \right] \\
&= \mathbf{E}_{X \sim \delta_{\mathcal{D}}(\cdot)} \left[\mathbf{E}_{Z_0 \sim q_{\phi_0}^0(\cdot), \gamma_0 \sim \mathcal{N}_\ell(\cdot)} [\log p_\theta(X, Z_K) \right. \\
&\quad \left. - \frac{1}{2} \rho_K^T M^{-1} \rho_K - \log q_{\phi_0}^0(Z_0) \right] + \frac{\ell}{2},
\end{aligned} \quad (12)$$

where we write $(Z_K, \rho_K) = \mathcal{H}_{\theta, \phi}(Z_0, (M/\beta_0)^{1/2} \gamma_0)$ under reparameterization. We can now consider the output of Algorithm 1 as taking a sample from the inner expectation for a given sample x from the outer expectation. A full algorithm to optimize (12) stochastically is provided in Algorithm 2. In practice, we take the gradients of (12) using automatic differentiation packages such as TensorFlow.

4 Experiments

In this section, we discuss the experiments used to validate our method. We first test HVAE on a simple tractable example (where no neural networks are needed), and then perform larger-scale tests on the MNIST dataset. We will provide code for all experiments online to foster reproducibility of experiments. All models were trained using TensorFlow [1].

Algorithm 2 Hamiltonian Variational Autoencoder

Require: $p_\theta(x, \cdot)$ is the unnormalized posterior for $x \in \mathcal{X}$ and $\theta \in \Theta$

```

function HVAE( $\mathcal{D}, K, n_B$ )                                     ▷  $n_B$  is minibatch size
  Initialize  $\theta, \phi$ 
  while  $\theta, \phi$  not converged do                               ▷ Stochastic optimization loop
    Sample  $\{x_1, \dots, x_{n_B}\} \sim \delta_{\mathcal{D}}(\cdot)$  independently
     $\hat{\mathcal{L}}_{\text{ELBO}}^H(\theta, \phi) \leftarrow 0$                              ▷ Want to average ELBO over the batch
    for  $i \leftarrow 1$  to  $n_B$  do
       $\hat{\mathcal{L}}_{\text{ELBO}}^H(\theta, \phi) \leftarrow \text{HIS}(x_i, \theta, K, \beta_0, \varepsilon, M) + \hat{\mathcal{L}}_{\text{ELBO}}^H(\theta, \phi)$ 
    ▷ Now, we perform gradient-based optimization updates according to RMSProp, SGA, etc.
     $\theta \leftarrow \text{UPDATETHETA}(\nabla_{\theta} \hat{\mathcal{L}}_{\text{ELBO}}^H(\theta, \phi) / n_B, \theta)$ 
     $\phi \leftarrow \text{UPDATEPHI}(\nabla_{\phi} \hat{\mathcal{L}}_{\text{ELBO}}^H(\theta, \phi) / n_B, \phi)$ 
  return  $\theta, \phi$ 

```

4.1 Gaussian Model

The generative model that we will consider first is a Gaussian likelihood with an offset and a Gaussian prior on the mean, given by

$$Z \sim \mathcal{N}(0, I_d),$$

$$x_i | Z \sim \mathcal{N}(Z + \Delta, \Sigma_X) \quad \text{independently,} \quad i \in [N]$$

where Σ_X is constrained to be diagonal. We will again write $\mathcal{D} \equiv \{x_1, \dots, x_N\}$ to denote an observed dataset under this model, where each $x_i \in \mathcal{X} \subseteq \mathbb{R}^d$. The goal of the problem is to learn the model parameters $\Sigma_X = \text{diag}(\sigma_1^2, \dots, \sigma_d^2)$ and $\Delta \in \mathbb{R}^d$, and calculate an estimate of the marginal log-likelihood $\log p_\theta(\mathcal{D})$ by importance sampling from the approximate posterior defined by our variational method. Note that in this case the true marginal log-likelihood is tractable, allowing us to compare estimates with the true values.

Here, the log unnormalized posterior is given by

$$\log p_\theta(\mathcal{D}, z) = \sum_{i=1}^N \log \mathcal{N}(x_i | z + \Delta, \Sigma_X) + \log \mathcal{N}(z | 0, I_d),$$

which can in this case be computed exactly using summary statistics so there is no need for the outer expectation with respect to $\delta_{\mathcal{D}}(\cdot)$ from (11) or (12).

For this example, we will use HVAE with variational prior being the true prior, i.e. $q_0 = p_0 = \mathcal{N}(\cdot | 0, I_d)$ and a fixed isotropic mass matrix $M = mI_d$. The gradient of the potential is given by

$$\nabla U_\theta(z | \mathcal{D}) = z + N \Sigma_X^{-1} (z + \Delta - \bar{x}),$$

where \bar{x} is the empirical mean of the dataset \mathcal{D} . The set of variational parameters here is $\phi_H \equiv \{\varepsilon, \beta_0\}$, where $\varepsilon \in \mathbb{R}^d$ contains the per-dimension leapfrog stepsizes and $\beta_0 \in (0, 1)$ is the initial inverse temperature. We constrain each of the leapfrog step sizes such that $\varepsilon_j \in (0, \xi)$ for some $\xi > 0$, for all $j \in [d]$ – this is to prevent the leapfrog discretization from entering unstable regimes. In practice, we simply set $\xi = 0.5$ without issue, although this can be lowered if instability issues arise in other settings, or alternatively raised if the step sizes approach this maximum in training without leapfrog displaying instability. Note that $\phi_H \in \mathbb{R}^{d+1}$ in this example; in particular, we do not optimize any parameters of the variational prior and thus require no further reparameterization.

We will compare HVAE with a basic Variational Bayes (VB) scheme with mean-field approximate posterior $q_{\phi_V}(z|x) = \mathcal{N}(z | \mu_Z, \Sigma_Z)$, where Σ_Z is diagonal and $\phi_V \equiv \{\mu_Z, \Sigma_Z\}$ denotes the set of learned variational parameters. This mirrors a typical VAE setup. Here, $\phi_V \in \mathbb{R}^{2d}$. We will also include a planar normalizing flow of the form of equation (10) in Rezende and Mohamed [20], but with the same flow parameters across iterations to ensure a fair comparison on the basis of the number of variational parameters. The variational prior q_0 is also set to the true prior p_0 as in HVAE above.

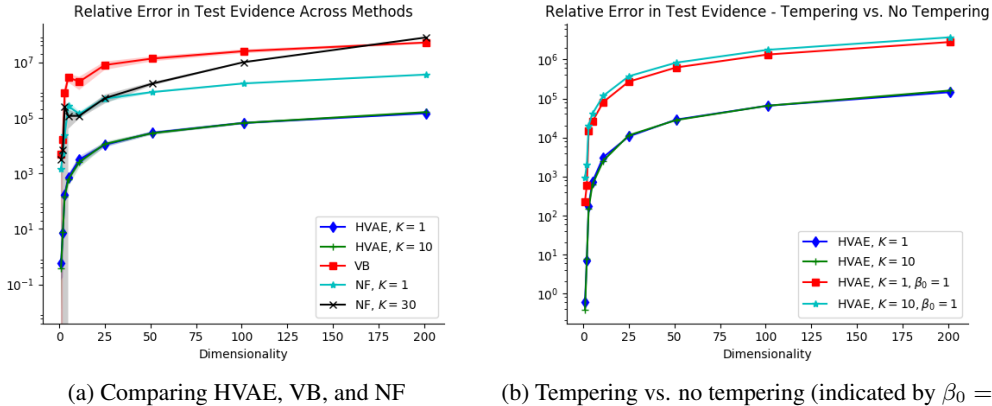


Figure 1: Averages of $|\log \hat{p}_\theta(\tilde{\mathcal{D}}) - \log p_\theta(\tilde{\mathcal{D}})|$ several variational methods and dimensions.

The log variational posterior $\log q_{\phi_N}(z|x)$ is given by equation (13) of Rezende and Mohamed [20], where $\phi_N \equiv \{\mathbf{u}, \mathbf{v}, b\}^2 \in \mathbb{R}^{2d+1}$.

We set our true offset vector to be $\Delta = (-\frac{d-1}{2}, \dots, \frac{d-1}{2})/5$, and our scale parameters to range quadratically from $\sigma_1 = 1$, reaching a minimum at $\sigma_{(d+1)/2} = 0.1$, and increasing back to $\sigma_d = 1$.³ All experiments have $N = 10,000$ and all training was done using RMSProp [25] with a learning rate of 10^{-3} .

To compare the results across methods, we will generate estimates of the log evidence $\log p_\theta(\tilde{\mathcal{D}})$ for out-of-sample datasets $\tilde{\mathcal{D}}$ using $L = 1000$ importance samples from the variational posteriors. We do this for 10 different and randomly-generated out-of-sample datasets for various choices of dimensionality d , displaying the average values of $|\log \hat{p}_\theta(\tilde{\mathcal{D}}) - \log p_\theta(\tilde{\mathcal{D}})|$, along with shaded error bands corresponding to one sample standard deviation, in Figure 1a. We notice that HVAE performs much better out of sample than the other methods, conceivably because the flow is guided by the true distribution. The error bands are also fairly tight across methods except in the lower-dimensional regime, statistically validating our claim that HVAE is doing better. We noticed that the VB scheme strongly overfit to the training data, leading to poor performance on the test data, while the NF schemes cannot really fit either the training or test data.

We also compare HVAE with tempering to HVAE without tempering, i.e. where β_0 is fixed to 1 in training – this has the effect of making our Hamiltonian dynamics homogeneous in time – again displaying the averages of $|\log \hat{p}_\theta(\tilde{\mathcal{D}}) - \log p_\theta(\tilde{\mathcal{D}})|$ across 10 out-of-sample datasets with $L = 1000$ importance samples in Figure 1b. We can clearly see that tempering and time-inhomogeneous dynamics are a key ingredient in the effectiveness of the method, which becomes even more apparent when we have a larger number of leapfrog steps.

4.2 Generative Model for MNIST

The next experiment that we consider is using HVAE to enhance a convolutional VAE for the binarized MNIST handwritten digit dataset. Again, our training data is $\mathcal{D} = \{x_1, \dots, x_N\}$, where each $x_i \in \mathcal{X} \subseteq \mathbb{R}^d$ for $d = 28 \times 28 = 784$. The base VAE has generative model $p_\theta(x_i, z_i) = p_\theta(x_i|z_i)p_0(z_i)$ for a single datapoint $x_i \in \mathcal{X}$, associated latent variable $z_i \in \mathbb{R}^\ell$, and model parameters $\theta \in \Theta$, independently for $i \in [N]$. The likelihood is given by a product of Bernoulli distributions: $p_\theta(x_i|z_i) = \prod_{j=1}^d \text{Bern}(\pi_\theta(z_i)_j)$, where $\pi_\theta(z_i) \in \mathbb{R}^d$ is the output of a convolutional neural network (the *generative network*, or *decoder*) parametrized by θ , and the prior p_0 is just a standard isotropic Gaussian on \mathbb{R}^ℓ . The variational approximate posterior is given by $q_{\phi_0}(z_i|x_i) = \mathcal{N}(z_i|\mu_{\phi_0}(x_i), \Sigma_{\phi_0}(x_i))$, where μ_{ϕ_0} and Σ_{ϕ_0} are separate outputs of the same neural network (the

²Boldface vectors used to match notation of Rezende and Mohamed [20].

³When d is even, $\sigma_{(d+1)/2}$ does not exist, although we could still consider $(d+1)/2$ to be the location of the minimum of the parabola defining the true standard deviations.

Table 1: Estimated Test Negative Log Likelihood on MNIST

	CNN VAE	HVAE, $K = 5$	HVAE, $K = 8$	HVAE, $K = 20$
Test NLL	84.982 ± 0.072	84.804 ± 0.042	84.786 ± 0.042	84.784 ± 0.045

inference network, or *encoder*) parametrized by the variational parameters ϕ_0 , and Σ_{ϕ_0} is constrained to be diagonal.

We attempted to match the network structure of Salimans et al. [23]. The inference network consists of three convolutional layers, each with filters of size 5×5 and a stride of 2. The convolutional layers output 16, 32, and 32 feature maps, respectively. The output of the third layer is fed into a fully-connected layer with hidden dimension $n_h = 300$, whose output is then fully connected to the output means and standard deviations each of size ℓ . Softplus activation functions are used throughout the network except immediately before the outputted mean. The generative network mirrors this structure in reverse, replacing the stride with upsampling as in Dosovitskiy et al. [7] and replicated in Salimans et al. [23].

We apply HVAE on top of the base convolutional VAE by considering the VAE variational approximate posterior as the *variational prior* to our method. We then evolve the samples from the variational prior according to Algorithm 1 and optimize the new objective given in (12). We reparameterize z_0 as $z_0 = \mu_{\phi_0}(x) + \Sigma_{\phi_0}(x)^{1/2}\epsilon$, for $\epsilon \sim \mathcal{N}(0, I_\ell)$ and $x \in \mathcal{X}$, to generate unbiased gradients of the ELBO with respect to ϕ_0 . We select various choices of K and set $\ell = 100$. In contrast with normalizing flows, we do not *amortize* our flow parameters ϵ and β_0 by allowing them to be outputs of the inference network, although this can also be incorporated into our framework. We use the standard stochastic binarization of MNIST [22] to train the model, and transition from RMSProp with learning rate 10^{-3} to stochastic gradient ascent with learning rate 10^{-5} throughout training for convergence. We calculate out-of-sample negative log likelihoods with $L = 1000$ importance samples and stochastic binarization, and report the average estimated negative log likelihood plus/minus the standard deviation in Table 1. We note that our base CNN VAE is not as powerful as the one in Salimans et al. [23] – implying some discrepancies in network structure which were present because their code was not available online. However, we are also able to improve upon the base model by adding our time-inhomogeneous Hamiltonian dynamics on top. Rezende and Mohamed [20] report only lower bounds on the log-likelihood for NFs, which are indeed lower than our log-likelihood estimates, although they use a much higher number of variational parameters.

5 Conclusion and Discussion

We have proposed a simple and principled way to exploit Hamiltonian dynamics within stochastic variational inference. Contrary to previous methods [23, 26], our algorithm does not rely on arbitrary reverse Markov kernels and benefits from the use of tempering ideas. Additionally, we can use the reparameterization trick to obtain an unbiased estimator of gradients of the ELBO. The resulting HVAE can be interpreted as a target-driven normalizing flow which requires the evaluation of a few gradients of the log-likelihood associated to a single data point at each stochastic gradient step. However, the Jacobian computations required for the ELBO are trivial. In our experiments, the robustness brought about by the use of target-informed dynamics can reduce the number of parameters that must be trained and improve generalizability.

There are many possible extensions of this approach. In particular, Hamiltonian dynamics preserves the Hamiltonian and hence also the corresponding target distribution, but there exist other deterministic dynamics which leave the target distribution invariant but not the Hamiltonian. This includes, in particular, the Nosé-Hoover thermostat. It is directly possible to use these dynamics instead of the Hamiltonian dynamics within the framework developed in subsection 2.3 and, in continuous-time, related ideas have appeared in physics [5, 19, 24]. This comes at the cost of slightly more complicated Jacobian calculations, although these can be simplified in certain regimes.

Another possibility is to allow the cooling parameters $\alpha_k \in (0, 1)$ to roam freely – i.e. not be constrained to a specific tempering schedule – and learn them variationally. This algorithmic improvement could also be combined with the standard *amortization* approach in which the variational flow parameters are outputs of the inference network, but we leave that for future work at this time.

References

- [1] Martín Abadi et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL <https://www.tensorflow.org/>. Software available from tensorflow.org.
- [2] Rianne van den Berg, Leonard Hasenclever, Jakub M Tomczak, and Max Welling. Sylvester normalizing flows for variational inference. *arXiv preprint arXiv:1803.05649*, 2018.
- [3] Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. In *The 4th International Conference on Learning Representations (ICLR)*, 2016.
- [4] Gavin E Crooks. Nonequilibrium measurements of free energy differences for microscopically reversible Markovian systems. *Journal of Statistical Physics*, 90(5-6):1481–1487, 1998.
- [5] Michel A Cuendet. Statistical mechanical derivation of Jarzynski’s identity for thermostated non-hamiltonian dynamics. *Physical Review Letters*, 96(12):120602, 2006.
- [6] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006.
- [7] Alexey Dosovitskiy, Jost Tobias Springenberg, and Thomas Brox. Learning to generate chairs with convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, pages 1538–1546. IEEE, 2015.
- [8] Paul Glasserman. *Gradient estimation via perturbation analysis*, volume 116. Springer Science & Business Media, 1991.
- [9] Matthew D Hoffman. Learning deep latent gaussian models with Markov chain Monte Carlo. In *International Conference on Machine Learning*, pages 1510–1519, 2017.
- [10] Matthew D Hoffman and Andrew Gelman. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.
- [11] Christopher Jarzynski. Nonequilibrium equality for free energy differences. *Physical Review Letters*, 78(14):2690, 1997.
- [12] Christopher Jarzynski. Hamiltonian derivation of a detailed fluctuation theorem. *Journal of Statistical Physics*, 98(1-2):77–102, 2000.
- [13] Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *The 2nd International Conference on Learning Representations (ICLR)*, 2014.
- [14] Diederik P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems*, pages 4743–4751, 2016.
- [15] Chris J Maddison, John Lawson, George Tucker, Nicolas Heess, Mohammad Norouzi, Andriy Mnih, Arnaud Doucet, and Yee Teh. Filtering variational objectives. In *Advances in Neural Information Processing Systems*, pages 6576–6586, 2017.
- [16] Radford M Neal. Annealed importance sampling. *Statistics and Computing*, 11(2):125–139, 2001.
- [17] Radford M Neal. Hamiltonian importance sampling. www.cs.toronto.edu/pub/radford/his-talk.ps, 2005. Talk presented at the Banff International Research Station (BIRS) workshop on Mathematical Issues in Molecular Dynamics.
- [18] Radford M Neal et al. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11), 2011.
- [19] Piero Procacci, Simone Marsili, Alessandro Barducci, Giorgio F Signorini, and Riccardo Chelli. Crooks equation for steered molecular dynamics using a Nosé-Hoover thermostat. *The Journal of Chemical Physics*, 125(16):164101, 2006.

- [20] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International Conference on Machine Learning*, pages 1530–1538, 2015.
- [21] Danilo Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *International Conference on Machine Learning*, pages 1278–1286, 2014.
- [22] Ruslan Salakhutdinov and Iain Murray. On the quantitative analysis of deep belief networks. In *Proceedings of the 25th international conference on Machine learning*, pages 872–879. ACM, 2008.
- [23] Tim Salimans, Diederik P Kingma, and Max Welling. Markov chain Monte Carlo and variational inference: Bridging the gap. In *International Conference on Machine Learning*, pages 1218–1226, 2015.
- [24] E Schöll-Paschinger and Christoph Dellago. A proof of Jarzynski’s nonequilibrium work theorem for dynamical systems that conserve the canonical distribution. *The Journal of Chemical Physics*, 125(5):054105, 2006.
- [25] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2): 26–31, 2012.
- [26] Christopher Wolf, Maximilian Karl, and Patrick van der Smagt. Variational inference with Hamiltonian Monte Carlo. *arXiv preprint arXiv:1609.08203*, 2016.