Coupled Variational Bayes via Optimization Embedding

*Bo Dai[1,2], *Hanjun Dai[1], Niao He[3], Weiyang Liu[1], Zhen Liu[1],
Jianshu Chen[4], Lin Xiao[5], Le Song[1,6]
[1]Georgia Institute of Technology, [2]Google Brain, [3]University of Illinois at Urbana Champaign
[4]Tencent AI, [5]Microsoft Research, [6]Ant Financial

Abstract

Variational inference plays a vital role in learning graphical models, especially on large-scale datasets. Much of its success depends on a proper choice of auxiliary distribution class for posterior approximation. However, how to pursue an auxiliary distribution class that achieves both good approximation ability and computation efficiency remains a core challenge. In this paper, we proposed *coupled variational Bayes* which exploits the *primal-dual view* of the ELBO with the variational distribution class generated by an optimization procedure, which is termed *optimization embedding*. This flexible function class couples the variational distribution with the original parameters in the graphical models, allowing end-to-end learning of the graphical models by back-propagation through the variational distribution. Theoretically, we establish an interesting connection to gradient flow and demonstrate the extreme flexibility of this implicit distribution family in the limit sense. Empirically, we demonstrate the effectiveness of the proposed method on multiple graphical models with either *continuous* or *discrete* latent variables comparing to state-of-the-art methods.

1 Introduction

Probabilistic models with Bayesian inference provides a powerful tool for modeling data with complex structure and capturing the uncertainty. The latent variables increase the flexibility of the models, while making the inference intractable. Typically, one resorts to approximate inference such as sampling [Neal, 1993, Neal et al., 2011, Doucet et al., 2001], or variational inference [Wainwright and Jordan, 2003, Minka, 2001]. Sampling algorithms enjoys good asymptotic theoretical properties, but they are also known to suffer from slow convergence especially for complex models. As a result, variational inference algorithms become more and more attractive, especially driven by the recent development on stochastic approximation methods [Hoffman et al., 2013].

Variational inference methods approximate the intractable posterior distributions by a family of distributions. Choosing a proper variational distribution family is one of the core problems in variational inference. For example, the mean-field approximation exploits the distributions generated by the independence assumption. Such assumption will reduce the computation complexity, however, it often leads to the distribution family that is too restricted to recover the exact posterior [Turner and Sahani, 2011]. Mixture models and nonparametric family [Jaakkola and Jordon, 1999, Gershman et al., 2012, Dai et al., 2016a] are the natural generalization. By introducing more components in the parametrization, the distribution family become more and more flexible, and the approximation error is reduced. However, the computational cost increases since it requires the evaluations of the log-likelihood and/or its derivatives for each component in each update, which could limit the scalability of variational inference. Inspired by the flexibility of deep neural networks, many neural networks parametrized distributions [Kingma and Welling, 2013, Mnih and Gregor, 2014]

*indicates equal contributions.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

and tractable flows [Rezende and Mohamed, 2015, Kingma et al., 2016, Tomczak and Welling, 2016, Dinh et al., 2016] have been introduced as alternative families in variational inference framework. The compromise in designing neural networks for computation tractability restricts the expressive ability of the approximation distribution family. Finally, the introduction of the variational distribution also brings extra separate parameters to be learned from data. As we know, the more flexible the approximation model is, the more samples are required for fitting such a model. Therefore, besides the approximation error and computational tractability, the sample efficiency should also be taken into account when designing the variational distribution family.

In summary, most existing works suffer from a trade-off between approximation accuracy, computation efficiency, and sample complexity. It remains open to design a variational inference approach that enjoys all three aspects. This paper provides a method towards such a solution, called *coupled variational Bayes* (CVB). The proposed approach hinges upon two key components: **i)**, the *primal-dual* view of the ELBO; and **ii)**, the *optimization embedding* technique for generating variational distributions. The primal-dual view of ELBO avoids the computation of determinant of Jacobian in flow-based model and makes the arbitrary flow parametrization applicable, therefore, reducing the approximation error. The optimization embedding generates an interesting class of variational distribution family, derived from the updating rule of an optimization procedure. This distribution class reduces separate parameters by coupling the variational distribution with the original parameters in the graphical models. Therefore, we can back-propagate the gradient w.r.t. the original parameters through the variational distributions, which promotes the sample efficiency of the learning procedure. We formally justify that in continuous-time case, such a technique implicitly provides a flexible enough approximation distribution family from the gradient flow view, implying that the CVB algorithm also guarantees zero approximation error in the limit sense. These advantages are further demonstrated in our numerical experiments.

In the remainder of this paper, we first provide a preliminary introduction to problem settings described in directed graphical models and the variational auto-encoder (VAE) framework in Section 2. We present our coupled variational Bayes in Section 3, which leverages the optimization embedding in the primal-dual view of ELBO to couple the variational distribution with original graphical models. We build up the connections of the proposed method with the existing flows formulations in Section 4. We demonstrate the empirical performances of the proposed algorithm in Section 5.

2 Background

Variational inference and learning Consider a probabilistic generative model, $p_\theta(x, z) = p_\theta(x|z)p(z)$, where $x \in \mathbb{R}^d$ denotes the observed variables and $z \in \mathbb{R}^r$ latent variables [2]. Given the dataset $\mathcal{D} = [x_i]_{i=1}^N$, one learns the parameter θ in the model by maximizing the marginal likelihood, *i.e.*, $\log \int p_\theta(x, z)dz$. However, the integral is intractable in general cases. Variational inference [Jordan et al., 1998] maximizes the evidence lower bound (ELBO) of the marginal likelihood by introducing an approximate posterior distribution, *i.e.*,

$$\log p_\theta(x) = \log \int p_\theta(x, z)dz \geqslant \mathbb{E}_{z\sim q_\phi(z|x)} \left[\log p_\theta(x, z) - \log q_\phi(z|x)\right], \tag{1}$$

where ϕ denotes the parameters of the variational distributions. There are two major issues in solving such optimization: **i)**, the appropriate parametrization for the introduced variational distributions, and **ii)**, the efficient algorithms for updating the parameters $\{\theta, \phi\}$. By adopting different variational distributions and exploiting different optimization algorithms, plenty of variants of variational inference and learning algorithms have been proposed. Among the existing algorithms, optimizing the objective with stochastic gradient descent [Hoffman et al., 2013, Titsias and Lázaro-gredilla, 2014, Dai et al., 2016a] becomes the dominated algorithm due to its scalability for large-scale datasets. However, how to select the variational distribution family has not been answered satifiedly yet.

Reparametrized density Kingma and Welling [2013], Mnih and Gregor [2014] exploit the recognition model or inference network to parametrize the variational distributions. A typical inference network is a stochastic mapping from the observation x to the latent variable z with a set of global parameters ϕ, *e.g.*, $q_\phi(z|x) := \mathcal{N}\left(z|\mu_{\phi_1}(x), \mathrm{diag}\left(\sigma^2_{\phi_2}(x)\right)\right)$, where $\mu_{\phi_1}(x)$ and $\sigma_{\phi_2}(x)$ are ofter

[2] We mainly discuss continuous latent variables in main text. However, the proposed algorithm can be extended to discrete latent variables easily as we show in Appendix B.

parametrized by deep neural networks. Practically, such reparameterizations have the closed-form of the entropy in general, and thus, the gradient computation and the optimization is relatively easy. However, such parameterization cannot perfectly fit the posterior when it does not fall into the known distirbution family, therefore, resulting extra approximation error to the true posterior.

Tractable flows-based model Parameterizing the variational distributions with *flows* is proposed to mitigate the limitation of expressive ability of the variational distribution. Specifically, assuming a series of invertible transformations as $\{\mathcal{T}_t : \mathbb{R}^r \to \mathbb{R}^r\}_{t=1}^T$ and $z^0 \sim q_0(z|x)$, we have $z^T = \mathcal{T}_T \circ \mathcal{T}_{T-1} \circ \ldots \circ \mathcal{T}_1\left(z^0\right)$ following the distribution $q_k(z|x) = q_0(z|x) \prod_{t=1}^T \left|\det \frac{\partial \mathcal{T}_t}{\partial z^t}\right|^{-1}$ by the change of variable formula. The flow-based parametrization generalizes the reparametrization tricks for the known distributions. However, a general parametrization of the transformation may violate the invertible requirement and result expensive or even infeasible calculation for the Jacobian and its determinant. Therefore, several carefully designed simple parametric forms of $\mathcal{T}$ have been proposed to compromise the invertible requirement and tractability of Jacobian [Rezende and Mohamed, 2015, Kingma et al., 2016, Tomczak and Welling, 2016, Dinh et al., 2016], at the expense of the flexibility of the corresponding variational distribution families.

3 Coupled Variational Bayes

In this section, we first consider the variational inference from a *primal-dual* view, by which we can avoid the computation of the determinant of the Jacobian. Then, we propose the *optimization embedding*, which generates the variational distribution by the adopt optimization algorithm. It automatically produces a *nonparametric* distribution class, which is flexible enough to approximate the posterior. More importantly, the optimization embedding couples the implicit variational distribution with the original graphical models, making the training more efficient. We introduce the key components below. Due to space limitation, we postpone the proof details of all the theorems in this section to Appendix A.

3.1 A Primal-Dual View of ELBO in Functional Space

As we introduced, the flow-based parametrization introduce more flexibility in representing the distributions. However, the calculating of the determinant of the Jacobian introduces extra computational cost and invertible requirement of the parametrization. In this section, we start from the primal-dual view perspective of ELBO, which will provide us a mechanism to avoid such computation and requirement, therefore, making the arbitrary flow parametrization applicable for inference.

As Zellner [1988], Dai et al. [2016a] show, when the family of variational distribution includes all valid distributions $\mathcal{P}$, the ELBO matches the marginal likelihood, *i.e.*,

$$L(\theta) := \mathbb{E}_{x\sim\mathcal{D}}\left[\log \int p_\theta(x,z)\,dz\right] = \max_{q(z|x)\in\mathcal{P}} \underbrace{\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{z\sim q(z|x)}\left[\log p_\theta(x|z) - KL\left(q(z|x)||p(z)\right)\right]}_{\ell_\theta(q)}, \tag{2}$$

where $p_\theta(x,z) = p_\theta(x|z)\,p(z)$ and $\mathbb{E}_{x\sim\mathcal{D}}[\cdot]$ denotes the expectation over empirical distribution on observations and $\ell_\theta(q)$ stands for the objective for the variational distribution in density space $\mathcal{P}$ under the probabilistic model with θ. Denote $q_\theta^*(z|x) := \operatorname{argmax}_{q(z|x)\in\mathcal{P}} \ell_\theta(q) = \frac{p_\theta(x,z)}{\int p_\theta(x,z)dz}$. The ultimate objective $L(\theta)$ will solely depend on θ, *i.e.*,

$$L(\theta) = \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{z\sim q_\theta^*(z|x)}\left[\log p_\theta(x,z) - \log q_\theta^*(z|x)\right], \tag{3}$$

which can be updated by stochastic gradient descent.

This would then require routinely solving the subproblem $\max_{q\in\mathcal{P}} \ell_\theta(q)$. Since the objective is taking over the whole distribution space, it is intractable in general. Traditionally, one may introduce special parametrization forms of distributions or flows for the sake of computational tractability, thus limiting the approximation ability. In what follows, we introduce an equivalent primal-dual view of the $\ell_\theta(q)$ in Theorem 1, which yields a promising opportunity to meet both approximation ability and computational tractability.

Theorem 1 (Equivalent reformulation of $L(\theta)$) *We can reformulate the $L(\theta)$ equivalently as*

$$\min_{\nu\in\mathcal{H}_+} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathbb{E}_{\xi\sim p_\xi(\cdot)}\left[\max_{z_{x,\xi}\in\mathbb{R}^r} \log p_\theta(x|z_{x,\xi}) - \log \nu(x, z_{x,\xi})\right] + \mathbb{E}_{z\sim p(z)}\left[\nu(x,z)\right]\right] - 1, \tag{4}$$

where $\mathcal{H}_+ = \left\{h : \mathbb{R}^d \times \mathbb{R}^r \to \mathbb{R}_+\right\}$, $p_\xi(\cdot)$ denotes some simple distribution and the optimal $\nu_\theta^*(x,z) = \frac{q_\theta^*(z|x)}{p(z)}$.

The primal-dual formulation of $L(\theta)$ is derived based on Fenchel-duality and interchangeability principle [Dai et al., 2016b, Shapiro et al., 2014]. With the primal-dual view of ELBO, we are able to represent the distributional operation on q by local variables $z_{x,\xi}$, which provides an implicit *nonparametric* transformation from $(x,\xi) \in \mathbb{R}^d \times \Xi$ to $z_{x,\xi} \in \mathbb{R}^p$. Meanwhile, with the help of dual function $\nu(x,z)$, we can also avoid the computation of the determinant of Jacobian matrix of the transformation, which is in general infeasible for arbitrary transformation.

3.2 Optimization Embedding

In this section, inspired by the local variable representation of the variational distribution in Theorem 1, we will construct a special variational distribution family, which integrates the variational distribution q, *i.e.*, transformation on local variables, and the original parameters of graphical models θ. We emphasize that optimization embedding is a general technique for representing the variational distributions and can also be accompanied with the original ELBO, which is provided in Appendix B.

As shown in Theorem 1, we switch handling the distribution $q(z|x) \in \mathcal{P}$ to each local variables. Specifically, given $x \sim \mathcal{D}$ and $\xi \sim p(\xi)$, with a fixed $\nu \in \mathcal{H}_+$,

$$z^*_{x,\xi;\theta} = \underset{z_{x,\xi}\in\mathbb{R}^p}{\operatorname{argmax}} \ \log p_\theta\left(x|z_{x,\xi}\right) - \log \nu\left(x, z_{x,\xi}\right). \tag{5}$$

For the complex graphical models, it is difficult to obtain the global optimum of (5). We can approach the $z^*_{x,\xi}$ by applying mirror descent algorithm (MDA) [Beck and Teboulle, 2003, Nemirovski et al., 2009]. Specifically, denote the initialization as $z^0_{x,\xi}$, in t-th iteration, we update the variables until converges via the prox-mapping operator

$$z^t_{x,\xi;\theta} = \underset{z\in\mathbb{R}^r}{\operatorname{argmax}} \ \left\langle z, \eta_t g\left(x, z^{t-1}_{x,\xi;\theta}\right)\right\rangle - D_\omega\left(z^{t-1}_{x,\xi;\theta}, z\right), \tag{6}$$

where $g\left(x, z^{t-1}_{x,\xi;\theta}\right) = \nabla_z \log p_\theta\left(x|z^{t-1}_{x,\xi;\theta}\right) - \nabla_z \log \nu\left(x, z^{t-1}_{x,\xi;\theta}\right)$ and $D_\omega(z_1,z_2) = \omega(z_2) - [\omega(z_1) + \langle \nabla\omega(z_1), z_2 - z_1\rangle]$ is the Bregman divergence generated by a continuous and strongly convex function $\omega(\cdot)$. In fact, we have the closed-form solution to the prox-mapping operator (6).

Theorem 2 (The closed-form of MDA) *Recall the $\omega(\cdot)$ is strongly convex, denote $f(\cdot) = \nabla\omega(\cdot)$, then, $f^{-1}(\cdot)$ exists. Therefore, the solution to* (6) *is*

$$z^t_{x,\xi;\theta} = f^{-1}\left(\eta_t g\left(x, z^{t-1}_{x,\xi;\theta}\right) + f\left(z^{t-1}_{x,\xi;\theta}\right)\right). \tag{7}$$

Proper choices of the Bregman divergences could exploit the geometry of the feasible domain and yield faster convergence. For example, if z lies in the general continuous space, one may use $\omega(z) = \frac{1}{2}\|z\|_2^2$, the $D_\omega(\cdot,\cdot)$ will be Euclidean distance on $\mathbb{R}^r$, $f(z) = z$ and $f^{-1}(z) = z$, and if z lies in a simplex, one may use $\omega(z) = \sum_{i=1}^r z_i \log z_i$, the $D_\omega(\cdot,\cdot)$ will be KL-divergence on the p-dim simplex, $f(z) = \log z$ and $f^{-1}(z) = \exp(z)$.

Assume we conduct the update (7) T iterations, the mirror descent algorithm outputs $z^T_{x,\xi;\theta}$ for each pair of (x,ξ). Therefore, it naturally establishes another *nonparametric* function that maps from $\mathbb{R}^d \times \Xi$ to $\mathbb{R}^r$ to approximate the sampler of the variational distribution point-wise, *i.e.*, $z^T_\theta(x,\xi) \approx z^*_{x,\xi;\theta}$, $\forall (x,\xi) \in \mathbb{R}^d \times \Xi$. Since such an approximation function is generated by the mirror descent algorithm, we name the corresponding function class as **optimization embedding**. Most importantly, the optimization embedded function explicitly depends on θ, which makes the end-to-end learning possible by back-propagation through the variational distribution. The detailed advantage of using the optimization embedding for learning will be explained in Section 3.3.

Before that, we first justify the approximation ability of the optimization embedding by connecting to the gradient flow for minimizing the KL-divergence with a special $\nu(x,z)$ in the limit case. For simplicity, we mainly focus on the basic case when $f(z) = z$. For a fixed x, sample $\xi \sim p(\xi)$, the particle $z^T_\theta(x,\xi)$ is recursively constructed by transform $\mathcal{T}_x(z) = z + \eta g(x,z)$. We show that

Theorem 3 (Optimization embedding as gradient flow) *For a continuous time $t = \eta T$ and infinitesimal step size $\eta \to 0$, the density of the particles $z^t \in \mathbb{R}^r$, denoted as $q_t(z|x)$, follows nonlinear Fokker-Planck equation*

$$\frac{\partial q_t(z|x)}{\partial t} = -\nabla\cdot\left(q_t(z|x)\, g_t(x,z)\right), \tag{8}$$

with $g_t(x,z) := \nabla_z \log p_\theta(x|z) - \nabla_z \log \nu^*_t(x,z)$ with $\nu^*_t(x,z) = \frac{q_t(z|x)}{p(z)}$. Such process defined by (8) is a gradient flow of KL-divergence in the space of measures with 2-Wasserstein metric.

Algorithm 1 Coupled Variational Bayes (CVB)

Initialize θ, V and W (the parameters of ν and z^0) randomly, set length of steps T and mirror function f.
Set $z^0(x,\xi) = h_W(x,\xi)$.
for iteration $k = 1, \ldots, K$ **do**
  Sample mini-batch $\{x_i\}_{i=1}^m$ from dataset $\mathcal{D}$, $\{z_i\}_{i=1}^m$ from prior $p(z)$, and $\{\xi_i\}_{i=1}^m$ from $p(\xi)$.
  for iteration $t = 1, \ldots, T$ **do**
    Compute $z_\theta^t(x,\xi)$ for each pair of $\{x_i, \xi_i\}_{i=1}^m$.
    Descend V with $\nabla_V \frac{1}{m}\sum_{i=1}^m \left[\nu_V(x_i, z_i) - \log \nu_V(x_i, z_\theta^t(x_i,\xi_i))\right]$.
  end for
  Ascend θ by stochastic gradient (11).
  Ascend W by $\nabla_W \frac{1}{m}\sum_{i=1}^m \left[\log p_\theta\left(x|z_\theta^T(x,\xi)\right) - \log \nu_V\left(x, z_\theta^T(x,\xi)\right)\right]$.
end for

From such a gradient flow view of optimization embedding, we can see that in limit case, the optimization embedding, $z_\theta^T(x,\xi)$, is flexible enough to approximate the posterior accurately.

3.3 Algorithm

Applying the optimization embedding into the $\ell_\theta(q)$, we arrive the approximate surrogate optimization to $L(\theta)$ in (2) as

$$\max_\theta \tilde{L}(\theta) := \min_{\nu\in\mathcal{H}_+} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathbb{E}_{\xi\sim p(\xi)}\left[\log p_\theta\left(x|z_\theta^T(x,\xi)\right) - \log \nu\left(x, z_\theta^T(x,\xi)\right)\right] + \mathbb{E}_{z\sim p(z)}\left[\nu(x,z)\right]\right]. \tag{9}$$

We can apply the stochastic gradient algorithm for (9) with the unbiased gradient estimator as follows.

Theorem 4 (Unbiased gradient estimator) *Denote*

$$\nu_\theta^*(x,z) = \operatorname*{argmin}_{\nu\in\mathcal{H}_+} \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{z\sim p(z)}\left[\nu(x,z)\right] - \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\log \nu\left(x, z_\theta^T(x,\xi)\right)\right], \tag{10}$$

we have the unbiased gradient estimator w.r.t. θ as

$$\begin{aligned}\frac{\partial \tilde{L}(\theta)}{\partial\theta} &= \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\frac{\partial \log p_\theta(x|z)}{\partial\theta}\Big|_{z=z_\theta^T(x,\xi)} + \frac{\partial \log p_\theta(x|z)}{\partial z}\Big|_{z=z_\theta^T(x,\xi)}\frac{\partial z_\theta^T(x,\xi)}{\partial\theta}\right] \\ &- \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\frac{\partial \log \nu_\theta^*(x,z)}{\partial z}\Big|_{z=z_\theta^T(x,\xi)}\frac{\partial z_\theta^T(x,\xi)}{\partial\theta}\right].\end{aligned} \tag{11}$$

As we can see from the gradient estimator (11), besides the effect on θ from the log-likelihood as in traditional VAE method with separate parameters of the variational distribution, which is the first term in (11), the estimator also considers the effect through the variational distribution explicitly in the second term. Such dependences through optimization embedding will potentially accelerate the learning in terms of sample complexity. The computation of the second term resembles to the back-propagation through time (BPTT) in learning the recurrent neural network, which can be easily implemented in Tensorflow or PyTorch.

Practical extension With the functional primal-dual view of ELBO and the optimization embedding, we are ready to derive the practical CVB algorithm. The CVB algorithm can be easily incorporated with parametrization into each component to balance among approximation flexibility, computational cost, and sample efficiency. The introduced parameters can be also trained by SGD within the CVB framework. For example, in the optimization embedding, the algorithm requires the initialization $z_{x,\xi}^0$. Besides the random initialization, we can also introduce a parametrized function for $z_{x,\xi}^0 = h_W(x,\xi)$, with W denoting the parameters. We can parametrize the $\nu(x,\xi)$ by deep neural networks with parameter V. To guarantee positive outputs of $\nu_V(x,\xi)$, we can use positive activation functions, *e.g.*, Gaussian, exponential, multi-quadratics, and so on, in the last layer. However, the neural networks parameterization may induce non-convexity, and thus, loss the guarantee of the global convergence in both (9) and (10), which leads to the bias in the estimator (11) and potential unstability in training. Empirically, to reduce the effect from neural network parametrization, we update the parameters in ν within the optimization embedding simultaneously, implicitly pushing z^T to follow the gradient flow. Taking into account of the introduced parameters, we have the CVB algorithm illustrated in Algorithm 1.

Moreover, we only discuss the optimization embedding through the basic mirror descent. In fact, other optimization algorithm, *e.g.*, the accelerated gradient descent, gradient descent with momentum, and other adaptive gradient method (Adagrad, RMSprop), can also be used for constructing the variational distributions. For the variants of CVB to parametrized continuous/discrete latent variables model and hybrid model with Langevin dynamics, please refer to the Appendix B and Appendix C.

4 Related Work

Connections to Langevin dynamics and Stein variational gradient descent As we show in Theorem 3, the optimization embedding could be viewed as a discretization of a nonlinear Fokker-Plank equation, which can be interpreted as a gradient flow of KL-divergence on 2-Wasserstein metric with a special $\nu(x, z)$. It resembles the gradient flow with Langevin dynamics [Otto, 2001]. However, Langevin dynamics is governed by a linear Fokker-Plank equation and results a stochastic update rule, *i.e.*, $z^t = z^{t-1} + \eta \nabla \log p_\theta \left(x, z^{t-1}\right) + 2\sqrt{\eta}\xi^{t-1}$ with $\xi^{t-1} \sim \mathcal{N}(0, 1)$, thus different from our deterministic update given the initialization z^0.

Similar to the optimization embedding, Stein variational gradient descent (SVGD) also exploits a nonlinear Fokker-Plank equation. However, these two gradient flows follow from different PDEs and correspond to different metric spaces, thus also resulting different deterministic updates. Unlike optimization embedding, the SVGD follows interactive updates between samples and requires to keep a fixed number of samples in the whole process.

Connection to adversarial variational Bayes (AVB) The AVB [Mescheder et al., 2017] can also exploit arbitrary flow and avoid the calculation related to the determinant of Jacobian via variational technique. Comparing to the primal-dual view of ELBO in CVB, AVB is derived based on classification density ratio estimation for KL-divergence in ELBO [Goodfellow et al., 2014, Sugiyama et al., 2012]. The most important difference is that CVB couples the adversarial component with original models through optimization embedding, which is flexible enough to approximate the true posterior and promote the learning sample efficiency.

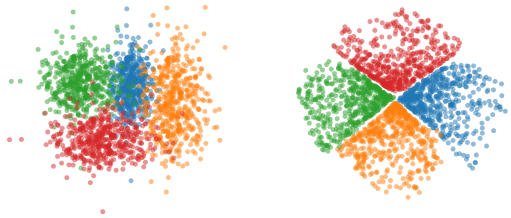
Connection to deep unfolding The optimization embedding is closely related to deep unfolding technique for inference and learning on graphical models. Existing scheme either unfold the *point estimation* through optimization [Domke, 2012, Hershey et al., 2014, Chen et al., 2015, Belanger et al., 2017, Chien and Lee, 2018], or expectation-maximization [Greff et al., 2017], or loopy belief propagation [Stoyanov et al., 2011]. In contrast, the we exploit optimization embedding through a flow pointwisely, so that it handles the distribution in a *nonparametric* way and ensures enough flexibility for approximation.

5 Experiments

In this section, we conduct empirical experiments to justify the benefits of the proposed coupled variational Bayes in terms of the flexibility and the efficiency in sample complexity. We also illustrate its generative ability. Additional experimental results, including the variants of CVB to discrete latent variable models and more results on real-world datasets, can be found in Appendix D. The implementation is released at `https://github.com/Hanjun-Dai/cvb`.

5.1 Flexibility in Posterior Approximation

We first justify the flexibility of the optimization embedding in CVB on the simple synthetic dataset [Mescheder et al., 2017]. It contains 4 data points, each representing a one-hot 2×2 binary image with non-zero entries at different positions. The generative model is a multivariate independent Bernoulli distribution with Gaussian distribution as prior, *i.e.*, $p_\theta(x|z) = \prod_{i=1}^{4} \pi_i(z)^{x_i}$ and $p(z) = \mathcal{N}(0, I)$ with $z \in \mathbb{R}^2$, and $\pi_i(z)$ is parametrized by 4-layer fully-connected neural networks with 64 hidden units in each latent layer. For CVB, we set $f(z) = z$ in optimization embedding. We emphasize the optimization embedding is non-parametric and generated automatically via mirror descent. The dual function $\nu(x, z)$ is parametrized



(1) vanilla VAE (2) CVB

Figure 1: Distribution of the latent variables for VAE and CVB on synthetic dataset.

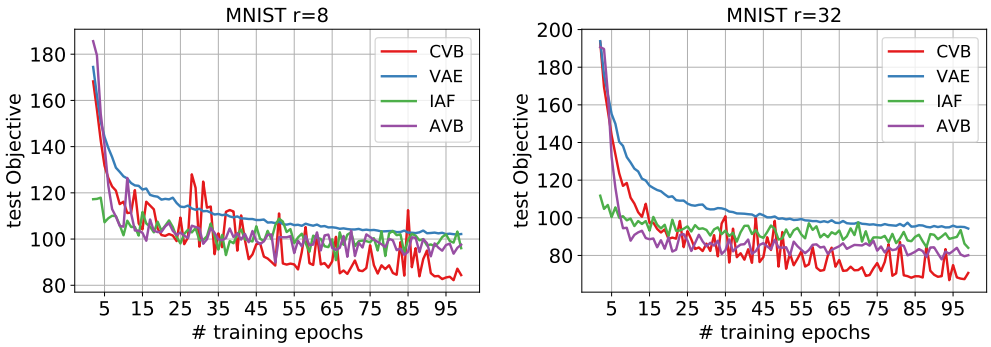


Figure 2: Convergence speed comparison in terms of number epoch on MNIST. We report the objective values of each method on held-out test set. The CVB achieves faster convergence speed comparing the other competitors in both $r = 8$ and $r = 32$ cases.

by a $(4 + 2)$-64-64-1 neural networks. The number of steps T in optimization embedding is set to be 5 in this case.

To demonstrate the flexibility of the optimization embedding, we compare the proposed CVB with the vanilla VAE with a diagonal Gaussian posterior. A separate encoder in VAE is parametrized by reversing the structure of the decoder. We visualize the obtained posterior by VAE and CVB in Figure 1. While VAE generates a mixture of 4 Gaussians that is consistent with the parametrization assumption, the proposed CVB divides the latent space with a complex distribution. Clearly, this yields that CVB is more flexible in terms of approximation ability.

5.2 Efficiency in Sample Complexity

To verify the sample efficiency of CVB, we compare the performance of CVB on static binarize MNIST dataset to the current state-of-the-art algorithms, including VAE with inverse autoregressive flow (VAE+IAF) [Kingma et al., 2016], adversarial variational Bayes (AVB) [Mescheder et al., 2017], and the vanilla VAE with Gaussian assumption for the posterior distribution (VAE) [Kingma and Welling, 2013]. In this experiment, we use the Gaussian as the initialization in CVB. We follow the same setting as AVB [Mescheder et al., 2017], where conditional generative model $P(x|z)$ is a Bernoulli that is parameterized with 3-layer convolutional neural networks (CNN), and the inference model is also a CNN which is parametrized reversely as the generative model. Experiments for AVB and VAE+IAF are conducted based on the codes provided by Mescheder et al. [2017][3], where the default neural network structure are adopted. For all the methods, in each epoch, the batch size is set to be 100 while the initial learning rate is set to 0.0001.

We illustrate the convergence speed of testing objective values in terms of number epoch in Figure 2. As we can see, in both cases with the dimension of latent variable $r = 8$ and $r = 32$, the proposed CVB, represented by the red curve, converges to a lower test objective value in a much faster speed. We also compare the final approximated log likelihood evaluated by Importance Sampling, with the best baseline results reported in the original papers in Table 1. In this case, the objective function becomes too optimistic about the actual likelihood. It could be caused by the Monte Carlo estimation of the Fenchel-Dual of KL-divergence, which is noisy comparing to the KL-divergence with closed-form in vanilla VAE. We can see that the proposed CVB still performs comparable with other alternatives. These results justify the benefits of parameters coupling through optimization embedding, especially in high dimension.

Table 1: The log-likelihood comparison between CVB and competitors on MNIST dataset. We can see that the proposed CVB achieves comparable performance on MNIST dataset.

Methods	$\log p(x) \approx$
CVB (8-dim)	-93.5
CVB (32-dim)	-84.0
AVB + AC (8-dim)	-89.6 [Mescheder et al., 2017]
AVB + AC (32-dim)	-80.2 [Mescheder et al., 2017]
DRAW + VGP	-79.9 [Tran et al., 2015]
VAE + IAF	-79.1 [Kingma et al., 2016]
VAE + NF ($T = 80$)	-85.1 [Rezende and Mohamed, 2015]
convVAE + HVI ($T = 16$)	-81.9 [Salimans et al., 2015]
VAE + HVI ($T = 16$)	-85.5 [Salimans et al., 2015]

[3]The code can be found on `https://github.com/LMescheder/AdversarialVariationalBayes`.

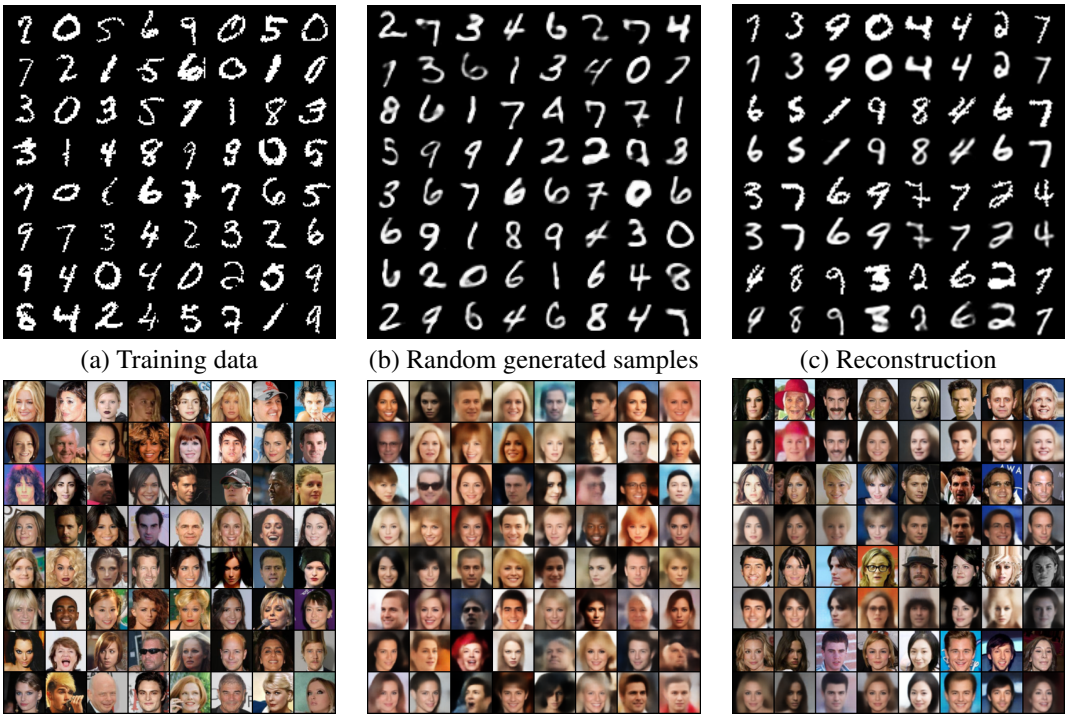
5.3 Generative Ability

We conduct experiments on real-world datasets, MNIST and CelebA, for demonstrating the generative ability of the model learned by CVB. For additional generated images, please refer to Appendix D.

MNIST We use the model that is specified in Section 5.2. The generated images and reconstructed images by the CVB learned model versus the training samples are illustrated in the first row of Figure 3.

CelebA We use CVB to train a generative model with deep deconvolution network on CelebA-dataset for a 64-dimension latent space with $\mathcal{N}(0, 1)$ prior [Mescheder et al., 2017]. we use convolutional neural network architecture similar to DCGAN. We illustrate the results in the second row of Figure 3.

We can see that the learned models can produces realistic images and reconstruct reasonably in both MNIST and CelebA datasets.



(a) Training data (b) Random generated samples (c) Reconstruction

(a) Training data (b) Random generated samples (c) Reconstruction

Figure 3: The training data, random generated images and the reconstructed images by the CVB learned models on MNIST and CelebA dataset. In the reconstruction column, the odd rows correspond to the test samples, and even rows correspond the reconstructed images.

6 Conclusion

We propose the coupled variational Bayes, which is designed based on the primal-dual view of ELBO and the optimization embedding technique. The primal-dual view of ELBO allows to bypass the difficulty with computing the Jacobian for non-invertible transformations and makes it possible to apply arbitrary transformation for variational inference. The optimization embedding technique, automatically generates a nonparametric variational distribution and couples it with the original parameters in generative models, which plays a key role in reducing the sample complexity. Numerical experiments demonstrates the superiority of CVB in approximate ability, computational efficiency, and sample complexity.

We believe the optimization embedding is an important and general technique, which is the first of the kind in literature and could be of independent interest. We provide several variants of the optimization embedding in Appendix B. It can also be applied to other models, *e.g.*, generative adversarial model and adversarial training, and deserves further investigation.

Acknowledgments

Part of this work was done when BD was with Georgia Tech. NH is supported in part by NSF CCF-1755829 and NSF CMMI-1761699. LS is supported in part by NSF IIS-1218749, NIH BIGDATA 1R01GM108341, NSF CAREER IIS-1350983, NSF IIS-1639792 EAGER, NSF IIS-1841351 EAGER, NSF CCF-1836822, NSF CNS-1704701, ONR N00014-15-1-2340, Intel ISTC, NVIDIA, Amazon AWS and Siemens.

References

Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

David Belanger, Bishan Yang, and Andrew McCallum. End-to-end learning for structured prediction energy networks. *arXiv preprint arXiv:1703.05667*, 2017.

D. P. Bertsekas. *Nonlinear Programming*. Athena Scientific, Belmont, MA, second edition, 1999.

Jianshu Chen, Ji He, Yelong Shen, Lin Xiao, Xiaodong He, Jianfeng Gao, Xinying Song, and Li Deng. End-to-end learning of lda by mirror-descent back propagation over a deep architecture. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1765–1773. 2015.

Jen-Tzung Chien and Chao-Hsi Lee. Deep unfolding for topic models. *IEEE transactions on pattern analysis and machine intelligence*, 40(2):318–331, 2018.

Bo Dai, Niao He, Hanjun Dai, and Le Song. Provable bayesian inference via particle mirror descent. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 985–994, 2016a.

Bo Dai, Niao He, Yunpeng Pan, Byron Boots, and Le Song. Learning from conditional distributions via dual embeddings. *CoRR*, abs/1607.04579, 2016b.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Justin Domke. Generic methods for optimization-based modeling. In *Artificial Intelligence and Statistics*, pages 318–326, 2012.

A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

Samuel Gershman, Matt Hoffman, and David M. Blei. Nonparametric variational inference. In John Langford and Joelle Pineau, editors, *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 663–670, New York, NY, USA, 2012. ACM.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

Klaus Greff, Sjoerd van Steenkiste, and Jürgen Schmidhuber. Neural expectation maximization. In *Advances in Neural Information Processing Systems*, pages 6694–6704, 2017.

John R Hershey, Jonathan Le Roux, and Felix Weninger. Deep unfolding: Model-based inspiration of novel deep architectures. *arXiv preprint arXiv:1409.2574*, 2014.

Matthew D. Hoffman, David M. Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347, 2013.

Tommi S. Jaakkola and Michael I. Jordon. Learning in graphical models. chapter Improving the Mean Field Approximation via the Use of Mixture Distributions, pages 163–173. MIT Press, Cambridge, MA, USA, 1999. ISBN 0-262-60032-3.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

M. I. Jordan, Z. Gharamani, T. S. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. In M. I. Jordan, editor, *Learning in Graphical Models*, pages 105–162. Kluwer Academic, 1998.

Yoon Kim, Sam Wiseman, Andrew C Miller, David Sontag, and Alexander M Rush. Semi-amortized variational autoencoders. *arXiv preprint arXiv:1802.02550*, 2018.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

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.

Qiang Liu. Stein variational gradient descent as gradient flow. In *Advances in neural information processing systems*, pages 3118–3126, 2017.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Joseph Marino, Yisong Yue, and Stephan Mandt. Iterative amortized inference. In *International Conference on Machine Learning*, pages 3400–3409, 2018.

Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. Adversarial variational bayes: Unifying variational autoencoders and generative adversarial networks. *arXiv preprint arXiv:1701.04722*, 2017.

T. Minka. *Expectation Propagation for approximative Bayesian inference*. PhD thesis, MIT Media Labs, Cambridge, USA, 2001.

Andriy Mnih and Karol Gregor. Neural variational inference and learning in belief networks. *arXiv preprint arXiv:1402.0030*, 2014.

Radford M. Neal. Probabilistic inference using Markov chain Monte Carlo methods. Technical report, Dept. of Computer Science, University of Toronto, 1993. CRG-TR-93-1.

Radford M Neal et al. Mcmc using hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11), 2011.

A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM J. on Optimization*, 19(4):1574–1609, January 2009. ISSN 1052-6234.

Felix Otto. The geometry of dissipative evolution equations: the porous medium equation. 2001.

Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*, 2015.

R. T. Rockafellar and R. J-B. Wets. *Variational Analysis*. Springer Verlag, 1998.

Tim Salimans, Diederik Kingma, and Max Welling. Markov chain monte carlo and variational inference: Bridging the gap. In *International Conference on Machine Learning*, pages 1218–1226, 2015.

Alexander Shapiro, Darinka Dentcheva, et al. *Lectures on stochastic programming: modeling and theory*, volume 16. SIAM, 2014.

Veselin Stoyanov, Alexander Ropson, and Jason Eisner. Empirical risk minimization of graphical model parameters given approximate inference, decoding, and model structure. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 725–733, 2011.

Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

Michalis Titsias and Miguel Lázaro-gredilla. Doubly stochastic variational bayes for non-conjugate inference. In Tony Jebara and Eric P. Xing, editors, *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1971–1979. JMLR Workshop and Conference Proceedings, 2014.

Jakub M Tomczak and Max Welling. Improving variational auto-encoders using householder flow. *arXiv preprint arXiv:1611.09630*, 2016.

Dustin Tran, Rajesh Ranganath, and David M Blei. The variational gaussian process. *arXiv preprint arXiv:1511.06499*, 2015.

Richard E Turner and Maneesh Sahani. Two problems with variational expectation maximisation for time-series models. *Bayesian Time series models*, 1(3.1):3–1, 2011.

M. J. Wainwright and M. I. Jordan. Graphical models, exponential families, and variational inference. Technical Report 649, UC Berkeley, Department of Statistics, September 2003.

Arnold Zellner. Optimal Information Processing and Bayes's Theorem. *The American Statistician*, 42(4), November 1988.

Appendix

A Proof of the Theorems in Section 3

We start by introducing the interchangeability principle Dai et al. [2016b], which plays a fundamental role for Theorem 1.

Lemma 5 (interchangeability principle Dai et al. [2016b]) *Let ξ be a random variable on Ξ and assume for any $\xi \in \Xi$, function $g(\cdot, \xi) : \mathbb{R} \to (-\infty, +\infty)$ is a proper*[4] *and upper semicontinuous*[5] *concave function. Then*

$$\mathbb{E}_\xi[\max_{u\in\mathbb{R}} g(u, \xi)] = \max_{u(\cdot)\in\mathcal{G}(\Xi)} \mathbb{E}_\xi[g(u(\xi), \xi)].$$

where $\mathcal{G}(\Xi) = \{u(\cdot) : \Xi \to \mathbb{R}\}$ is the entire space of functions defined on support Ξ.

The result implies that one can replace the expected value of point-wise optima by the optimum value over a function space. More general results of interchange between maximization and integration can be found in [Rockafellar and Wets, 1998, Chapter 14] and [Shapiro et al., 2014, Chapter 7].

Proof of Theorem 1: We apply the Fenchel dual form of KL-divergence, we have

$$KL\left(q||p\right) = \left\langle q, \log\frac{q}{p}\right\rangle = \max_{\nu>0} \left\langle q, \log\nu\right\rangle - \left\langle p, \nu\right\rangle + 1,$$

and

$$\nu^* = \operatorname*{argmax}_{\nu>0} \left\langle q, \log\nu\right\rangle - \left\langle p, \nu\right\rangle + 1 = \frac{q}{p}.$$

In fact, these equations are easy to verify by taking the gradient of the objective function and setting to zero. Plug such variational form into $\ell_\theta\left(q\right)$, we have

$$L\left(\theta\right) = \max_{q\in\mathcal{P}} \min_{\nu\in\mathcal{H}_+} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathbb{E}_q\left[\log p_\theta(x|z) - \log\nu(x,z)\right] + \mathbb{E}_{z\sim p(z)}\left[\nu(x,z)\right]\right] - 1, \tag{12}$$

where $\mathcal{H}_+$ denotes the space which contains all positive functions, *i.e.*, $\mathcal{H}_+ = \left\{h : \mathbb{R}^d \times \mathbb{R}^r \to \mathbb{R}_+\right\}$. It is easy to verify (12) is concave-convex, therefore, the strong duality holds, which implies,

$$\begin{aligned} L\left(\theta\right) &= \min_{\nu\in\mathcal{H}_+} \max_{q\in\mathcal{P}} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathbb{E}_q\left[\log p_\theta(x|z) - \log\nu(x,z)\right] + \mathbb{E}_{z\sim p(z)}\left[\nu\left(x,z\right)\right]\right] - 1 \\ &= \min_{\nu\in\mathcal{H}_+} \max_{z(x,\xi)\in\mathcal{F}} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathbb{E}_{\xi\sim p(\xi)}\left[\log p_\theta\left(x|z\left(x,\xi\right)\right) - \log\nu\left(x, z\left(x,\xi\right)\right)\right] + \mathbb{E}_{z\sim p(z)}\left[\nu(x,z)\right]\right] - 1, \end{aligned}$$

where the second equality comes from reparametrization and $\mathcal{F}$ denotes the transport mapping function space. In other words, as long as the $\mathcal{F}$ is flexible enough so that containing the function $z^*\left(x,\cdot\right)$ transform $p(\xi)$ to $q^*(z|x;\theta)$, the equality holds.

Under the mild condition that $\log p_\theta(x|\cdot)$ and $\log\nu\left(x,\cdot\right)$ are continuous, by applying the interchangeable principle in Lemma 5, we arrive the conclusion, *i.e.*,

$$L\left(\theta\right) = \min_{\nu\in\mathcal{H}_+} \mathbb{E}_{x\sim\mathcal{D}}\left[\mathbb{E}_{\xi\sim p(\xi)}\left[\max_{z_{x,\xi}\in\mathbb{R}^r} \log p_\theta\left(x|z_{x,\xi}\right) - \log\nu\left(x, z_{x,\xi}\right)\right] + \mathbb{E}_{z\sim p(z)}\left[\nu(x,z)\right]\right] - 1.$$

■

Proof of Theorem 2: We take the derivative to (6) w.r.t. z and set to zero, resulting

$$\begin{aligned} &\eta_t g\left(x, z^{t-1}_{x,\xi;\theta}\right) - f\left(z\right) + f\left(z^{t-1}_{x,\xi;\theta}\right) = 0 \\ \Rightarrow\quad & z^t_{x,\xi;\theta} = f^{-1}\left(\eta_t g\left(x, z^{t-1}_{x,\xi;\theta}\right) + f\left(z^{t-1}_{x,\xi;\theta}\right)\right). \end{aligned}$$

The f^{-1} exists due to the property of the strongly convexity of $\omega\left(\cdot\right)$.

■

[4] We say $g(\cdot,\xi)$ is proper when $\{u \in \mathbb{R} : g(u,\xi) < \infty\}$ is non-empty and $g(u,\xi) > -\infty$ for $\forall u$.

[5] We say $g(\cdot,\xi)$ is upper semicontinuous when $\{u \in \mathbb{R} : g(u,\xi) < \alpha\}$ is an open set for $\forall\alpha \in \mathbb{R}$. Similarly, we say $g(\cdot,\xi)$ is lower semicontinuous when $\{u \in \mathbb{R} : g(u,\xi) > \alpha\}$ is an open set for $\forall\alpha \in \mathbb{R}$.

To prove Theorem 3, we first need the lemma Lemma 6, proved by Liu [2017], to guarantee the inconvertible of the transform.

Lemma 6 ([Liu, 2017]) *Let B be a square matrix and $\|B\|_F$ be the Forbenius norm. Let ϵ be a positive number such that $0 \leqslant \epsilon \leqslant \frac{1}{\rho(B+B^\top)}$, where $\rho(\cdot)$ denotes the spectrum radius. Then, $I + \epsilon\left(B + B^\top\right)$ is positive definite, and*

$$\log|\det(I+\epsilon B)| \geqslant \epsilon \operatorname{tr}(B) - \epsilon^2 \frac{\|B\|_F^2}{1-\epsilon\rho\left(B+B^\top\right)}.$$

Therefore, take an even small ϵ such that $0 \leqslant \epsilon \leqslant \frac{1}{2\rho(B+B^\top)}$, we get

$$\log|\det(I+\epsilon B)| \geqslant \epsilon \operatorname{tr}(B) - 2\epsilon^2 \|B\|_F^2.$$

Proof of Theorem 3: The conclusion can be obtained by directly applying the Fokker-Planck Equation. We prove the result by infinitesimal analysis similar to Liu [2017][Appendix A.3].

For a fixed x, recall $\mathcal{T}_x(z) = z + \eta g(x,z)$, we denote $z \sim q(z|x)$. With a sufficient small η, $\nabla\mathcal{T}_x(z) = I + \eta\nabla g(x,z)$ is positive definite by lemma Lemma 6. Therefore, we have the inverse function of $\mathcal{T}_x^{-1}(z)$ as

$$\mathcal{T}_x^{-1}(z) = z - \eta g(x,z) + o(\eta).$$

The density of $\mathcal{T}_x(z)$ can be calculated by change of variables formula,

$$q'(z|x) = q\left(\mathcal{T}_x^{-1}(z)|x\right) \cdot \left|\det\left(\nabla\mathcal{T}_x^{-1}(z)\right)\right|.$$

Then, we have

$$\begin{aligned}
\log q'(z|x) &= \log q\left(\mathcal{T}_x^{-1}(z)|x\right) + \log\left|\det\left(\nabla\mathcal{T}_x^{-1}(z)\right)\right| \\
&= \log q(z - \eta g(x,z)|x) + \log\det(z - \eta\nabla_z g(x,z) + o(\eta)) + o(\eta) \\
&= \log q(z|x) - \eta\nabla_z \log q(z|x)^\top g(x,z) - \eta\operatorname{tr}(\nabla_z g(x,z)) + o(\eta)
\end{aligned}$$

where the third equation comes from Taylor expansion.

Therefore, by the definition of the derivative of $\log(\cdot)$, we have

$$\begin{aligned}
\frac{q'(z|x) - q(z|x)}{\eta} &= \frac{q(z|x)\left(\log q'(z|x) - \log q(z|x)\right)}{\eta} + o(\eta) \\
&= -q(z|x)\left(\eta\nabla_z \log q(z|x)^\top g(x,z) + \eta\operatorname{tr}(\nabla_z g(x,z))\right) + o(\eta) \\
&= -\nabla\cdot(q(z|x)\, g(x,z)) + o(\eta),
\end{aligned}$$

which results the PDE as

$$\frac{\partial q_t(z|x)}{\partial t} = -\nabla\cdot(q_t(z|x)\, g_t(x,z)).$$

Recall $\nu_t(x,z) = \frac{q_t(z|x)}{p(z)}$ as proved in Theorem 1, we have the PDE as

$$\begin{aligned}
\frac{\partial q_t(z|x)}{\partial t} &= -\nabla\cdot\left(q_t(z|x)\nabla\log\frac{p_\theta(x,z)}{q_t(z|x)}\right) \\
&= -\nabla\cdot(q_t(z|x)\nabla\log p_\theta(x,z)) + \Delta q_t(z|x) \\
&= -\frac{dKL\left(q_t(z|x)\,||p_\theta(x,z)\right)}{dt} \\
&= \left\|\nabla\log\left(\frac{q_t(z|x)}{p_\theta(x,z)}\right)\right\|_{L_{q_t}}^2.
\end{aligned}$$

Therefore, the PDE can be viewed as a gradient flow of KL-divergence under 2-Wasserstein metric [Otto, 2001].

■

Proof of Theorem 4: The gradient estimator (11) can be directly obtained by applying the chain-rule with Danskin's theorem [Bertsekas, 1999]. We provide the derivation below for completeness.

$$\begin{aligned}\frac{\partial \tilde{L}(\theta)}{\partial \theta} &= \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\frac{\partial \log p_\theta(x|z)}{\partial\theta}\Big|_{z=z_\theta^T(x,\xi)} + \frac{\partial \log p_\theta(x|z)}{\partial z}\Big|_{z=z_\theta^T(x,\xi)}\frac{\partial z_\theta^T(x,\xi)}{\partial\theta}\right] \\ &- \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\frac{\partial \log \nu_\theta^*(x,z)}{\partial z}\Big|_{z=z_\theta^T(x,\xi)}\frac{\partial z_\theta^T(x,\xi)}{\partial\theta}\right] \\ &+ \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{z\sim p(z)}\left[\frac{\partial \nu_\theta^*(x,z)}{\partial\theta}\right] - \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\frac{\partial \log \nu_\theta^*(x,z)}{\partial\theta}\Big|_{z=z_\theta^T(x,\xi)}\right].\end{aligned}$$

Denote $l(\theta) = \min_{\nu\in\mathcal{H}_+}\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{z\sim p(z)}[\nu(x,z)] - \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\log\nu\left(x, z_\theta^T(x,\xi)\right)\right]$, we can rewrite the third term as

$$\frac{\partial l(\theta)}{\partial\nu}\Big|_{\nu=\nu_\theta^*}\frac{\partial\nu}{\partial\theta} = \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{z\sim p(z)}\left[\frac{\partial\nu_\theta^*(x,z)}{\partial\theta}\right] - \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\frac{\partial\log\nu_\theta^*(x,z)}{\partial\theta}\Big|_{z=z_\theta^T(x,\xi)}\right].$$

Recall the optimality of $\nu_\theta^*(x,z)$, it is easy to verify that $\frac{\partial l(\theta)}{\partial\nu}\Big|_{\nu=\nu_\theta^*} = 0$, and thus, the third term in $\frac{\partial\tilde{L}(\theta)}{\partial\theta}$ is zero.

■

B Several Variants of CVB

In Section 3, we mainly discussed the most general setting for $z_\theta^T(x,\xi)$, *i.e.*, we conduct optimization embedding for each pair of $(x,\xi)\in\mathbb{R}^d\times\Xi$ without any disributional form assumption. This provides the most flexible family for the variational distribution with the extra cost in fitting the dual function $\nu_V(x,z)$. In this section, we show several variants of CVB which are derived from applying optimization embedding to the posteriors for each pair with pre-fixed density forms, including Gaussian, categorical, and flow-based distributions in Section B.1, Section B.2, and Section B.3, respectively. In other words, we extend the optimization embedding under particular distribution assumption for *each* $q(z|x)$. We emphasize that it is still *nonparametric* since for each x, it owns a *separate* posterior. It is different from the vanilla amortized inference in VAE where the posterior parametrization are shared across all the samples. As we will see, these variants of CVB for pre-fixed parametric variational family will lead to Kim et al. [2018], Marino et al. [2018] as special cases of the framework. Finally, we apply the general optimization embedding technique to the parameters in the vanilla amortized VAE in Section B.4, resulting the parametric CVB.

These variants of CVB sacrifice the approximate ability for better computational efficiency, while still keep better sample complexity. In summary, we have the variants of CVB illustrated in Algorithm 2.

Algorithm 2 CVB for Parametric Variational Posterior

1: Initialize θ and W randomly, set length of steps T and mirror update f.
2: **for** epoch $k = 1,\ldots,K$ **do**
3: Sample mini-batch $\{x_i\}_{i=1}^m$ from dataset $\mathcal{D}$ and $\{\xi_i\}_{i=1}^m$ from $\mathcal{N}(0,\mathbf{I})$.
4: Compute the $\left\{z_\theta^T(x_i,\xi_i)\right\}_{i=1}^m$ via (16) for Gaussian latent variables, or via (22) for categorical latent variables, or via (27) for flow-based latent variables.
5: Update θ and W by stochastic gradient **ascend** with corresponding gradient estimator $\nabla_\theta\tilde{L}(\theta,W)$ and $\nabla_W\tilde{L}(\theta,W)$.
6: **end for**

B.1 Optimization Embedding for Gaussian Latent Variables

We first illustrate applying the optimization embedding for continuous latent variables whose posterior is assumed as Gaussian, *i.e.*, $q(z|x) = \mathcal{N}\left(z|\phi_x, \mathrm{diag}\left(\psi_x^2\right)\right)$ where $\{\phi_x,\psi_x\}$ denote the parameters depend on x. Therefore, we have $z(x,\xi) = \phi_x + \psi_x\cdot\xi$ with $\xi\sim\mathcal{N}(0,\mathbf{I})$. With such parametrization, the KL-divergence term in the ELBO will have closed-form, therefore, we do not need to introduce

the dual function $\nu(x,z)$. The EBLO, $L(\theta)$ becomes

$$\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim\mathcal{N}(0,\mathbf{I})}\left[\max_{\phi_x,\psi_x}\underbrace{\log p_\theta\left(x|\phi_x+\psi_x\cdot\xi\right)+\frac{1}{2}\cdot\mathbf{1}^\top\left(2\log\psi_x-\phi_x^2-\psi_x^2\right)}_{\ell_\theta(\phi_x,\psi_x)}\right]. \tag{13}$$

Then, we can embed the optimization algorithm for ϕ_x,ψ_x to build up the connection between $z(x,\xi)$ with θ. Specifically, we write out the updates for ϕ_x and ψ_x,

$$\phi_{x,\theta}^{t+1}=f^{-1}\left(\eta_t g_{\phi_x^t}\left(x,\theta\right)+f\left(\phi_{x,\theta}^t\right)\right), \tag{14}$$

$$\psi_{x,\theta}^{t+1}=f^{-1}\left(\eta_t g_{\psi_x^t}\left(x,\theta\right)+f\left(\psi_{x,\theta}^t\right)\right), \tag{15}$$

where $g_{\phi_x^t}(x,\theta)$ and $g_{\psi_x^t}(x,\theta)$ denote the gradient of $\ell_\theta(\phi_x,\psi_x)$ w.r.t. ϕ_x and ψ_x, respectively. We can also initialize the $\left[\phi_{x,\theta}^0,\psi_{x,\theta}^0\right]=h_W(x)$, where W will be learned together by SGD. Therefore, after T steps of the iteration, we have the function $z_\theta^T(x,\xi)$ from $\mathbb{R}^d\times\Xi$ to $\mathbb{R}^p$ as

$$z_\theta^T(x,\xi)=\phi_{x,\theta}^T+\psi_{x,\theta}^T\cdot\xi,\quad \xi\sim\mathcal{N}(0,\mathbf{I}). \tag{16}$$

Comparing the (16) with the most general (7), although $z_\theta^T(x,\xi)$ is still nonparametric in the sense z changes individually for each pair (x,ξ), the effect of the parametrization of posterior restricts the $z_\theta^T(x,\xi)$ to be a special form as derived in (16).

Plug (16) into $L(\theta)$, we have the surrogate objective $\tilde{L}(\theta,W)$ defined as

$$\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim\mathcal{N}(0,\mathbf{I})}\left[\log p_\theta\left(x|z_\theta^T(x,\xi)\right)+\frac{1}{2}\cdot\mathbf{1}^\top\left(2\log\psi_{x,\theta}^T-\left(\phi_{x,\theta}^T\right)^2-\left(\psi_{x,\theta}^T\right)\right)^2\right].$$

Notice that in $\tilde{L}(\theta,W)$, we use optimization embedding cancel the $\max$-operator on $\{\phi_x,\psi_x\}$. More importantly, we explicitly couple these parameters in the variational distributions with the parameter θ in the generative model. Then, we can apply the SGD for learning the θ and W. Similar to Theorem 4, we can derive the gradient estimator of θ as

$$\begin{aligned}\frac{\partial\tilde{L}(\theta,W)}{\partial\theta} &= \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim\mathcal{N}(0,\mathbf{I})}\left[\frac{\partial\log p_\theta(x|z)}{\partial\theta}\Big|_{z=z_\theta^T(x,\xi)}+\frac{\partial\log p_\theta(x|z)}{\partial z}\Big|_{z=z_\theta^T(x,\xi)}\frac{\partial z_\theta^T(x,\xi)}{\partial\theta}\right]\\ &+ \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim\mathcal{N}(0,\mathbf{I})}\left[\frac{1}{2}\cdot\mathbf{1}^\top\left(\left(\frac{2}{\psi_{x,\theta}^T}-2\psi_{x,\theta}^T\right)\frac{\partial\psi_{x,\theta}^T}{\partial\theta}-2\phi_{x,\theta}^T\frac{\partial\phi_{x,\theta}^T}{\partial\theta}\right)\right]\end{aligned} \tag{17}$$

As we can see, the first term in (17) and (11) is the same, while the second term is different. Due to the closed-form of KL-divergence in Gaussian parametrization, we can calculate the gradient w.r.t. θ of the KL-divergence term in ELBO directly as in (17), while in the general case (11), we need to fit the $\nu(x,\xi)$ for approximating the KL-divergence.

B.2 Optimization Embedding for Categorical Latent Variables

Similarly, the optimization embedding can also be applied to categorical latent variables models. To ensure the gradient is valid, we approximate the categorical latent variables with Gumbel-Softmax [Jang et al., 2016, Maddison et al., 2016], *i.e.*,

$$q_\phi(z|x)=\Gamma(r)\,\tau^{r-1}\left(\sum_{i=1}^r\frac{\pi_{x,\phi,i}}{z_i^\tau}\right)^{-r}\prod_{i=1}^r\left(\frac{\pi_{x,\phi,i}}{z_i^{\tau+1}}\right), \tag{18}$$

and

$$z_{\phi,i}(x,\xi)=\frac{\exp\left(\left(\phi_{x,i}+\xi_i\right)/\tau\right)}{\sum_{i=1}^r\exp\left(\left(\phi_{x,i}+\xi_i\right)/\tau\right)},\quad \xi_i\sim\mathcal{G}(0,1),\quad i\in\{1,\ldots,r\}, \tag{19}$$

with $\pi_{x,\phi,i} = \frac{\exp(\phi_{x,i})}{\sum_{i=1}^{p}\exp(\phi_{x,i})}$ and $\mathcal{G}(0,1)$ denotes the Gumbel distribution. Denote the initialization function parametrized by W, follow the same derivation, we have the ELBO, $L(\theta)$, as

$$\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim\mathcal{G}(0,\mathbf{I})}\left[\max_{\phi_x}\underbrace{\log p_\theta\left(x, z_\phi\left(x,\xi\right)\right) + p\log\left(\mathbf{1}^\top\frac{\pi_{x,\phi}}{\left(z_\phi\left(x,\xi\right)\right)^\tau}\right) - \mathbf{1}^\top\log\frac{\pi_{x,\phi}}{\left(z_\phi\left(x,\xi\right)\right)^{\tau+1}}}_{\ell_\theta(\phi_x)}\right]. \tag{20}$$

Similarly, we embed the optimization procedure for ϕ_x, resulting

$$\phi_{x,\theta}^{t+1} = f^{-1}\left(\eta_t g_{\phi_{x,\theta}^t}\left(x,\theta\right) + f\left(\phi_{x,\theta}^t\right)\right), \tag{21}$$

$g_{\phi_{x,\theta}^t}\left(x,\theta\right)$ denotes the gradient of $\ell_\theta\left(\phi_x\right)$ w.r.t. ϕ. Therefore, after T steps of the iteration, we have the function $z_\theta^T\left(x,\xi\right)$ from $\mathbb{R}^d\times\Xi$ to $\mathbb{R}^r$ as

$$z_{\theta,i}^T\left(x,\xi\right) = \frac{\exp\left(\left(\phi_{x,\theta,i}^T + \xi_i\right)/\tau\right)}{\sum_{i=1}^{p}\exp\left(\left(\phi_{x,\theta,i}^T + \xi_i\right)/\tau\right)}, \quad \xi_i\sim\mathcal{G}\left(0,1\right), \quad i\in\left\{1,\ldots,r\right\}, \tag{22}$$

where $\phi_{x,\theta}^0 = h_W\left(x\right)$. Plug (22) into $L\left(\theta\right)$, we have the surrogate objective $\tilde{L}\left(\theta\right)$ defined as

$$\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim\mathcal{G}(0,\mathbf{I})}\left[\log p_\theta\left(x, z_\theta^T\left(x,\xi\right)\right) + p\log\left(\mathbf{1}^\top\frac{\pi_{\phi_{x,\theta}^T}}{\left(z_\theta^T\left(x,\xi\right)\right)^\tau}\right) - \mathbf{1}^\top\log\frac{\pi_{\phi_{x,\theta}^T}}{\left(z_\theta^T\left(x,\xi\right)\right)^{\tau+1}}\right].$$

We can follow the chain-rule to calculate the gradient estimator w.r.t. θ and W for $\tilde{L}\left(\theta,W\right)$, and apply the SGD to optimize $\tilde{L}\left(\theta,W\right)$ w.r.t. both θ and W.

B.3 Optimization Embedding for Flow-based Latent Variables

In this section, we derive the optimization embedding for the distributions generated by change-of-variables, *i.e.*, flow-based models. Specifically, we assume that the latent variables follow the distribution generated by change of variables, *i.e.*,

$$q_\phi\left(z|x\right) = p\left(\mu_{\phi_x}^{-1}\left(z\right)\right)\left|\det\frac{\partial\mu_{\phi_x}^{-1}\left(z\right)}{\partial z}\right|. \tag{23}$$

where $\mu_{\phi_x}\left(\cdot\right)$ denotes the bijective function and $p\left(\cdot\right)$ denotes some simple distribution. Then, we can sample z as

$$z_\phi\left(x,\xi\right) = \mu_{\phi_x}\left(\xi\right), \quad \xi\sim p\left(\xi\right). \tag{24}$$

It is easy to see that the Gaussian and categorical distributed latent variable are just special cases of the general change-of-variables. There are several carefully designed simple parametric forms of μ_{ϕ_x} have been proposed to compromise the invertible requirement and tractability of Jacobian, *e.g.*, normalizing flow [Rezende and Mohamed, 2015, Tomczak and Welling, 2016], autoregressive flow [Kingma et al., 2016], and partition flow [Dinh et al., 2016]. The optimization embedding can be applied to all of these parametrizations. Follow the same derivation, we have the ELBO, $L\left(\theta\right)$, as

$$\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\max_{\phi_x}\underbrace{\log p_\theta\left(x,\mu_{\phi_x}\left(\xi\right)\right) + \log p\left(\xi\right) - \log\left|\det\frac{\partial\mu_{\phi_x}\left(\xi\right)}{\partial\xi}\right|}_{\ell_\theta(\phi_x)}\right]. \tag{25}$$

Similarly, we embed the optimization procedure for the ϕ_x, resulting

$$\phi_{x,\theta}^{t+1} = f^{-1}\left(\eta_t g_{\phi_{x,\theta}^t}\left(x,\theta\right) + f\left(\phi_{x,\theta}^t\right)\right), \tag{26}$$

$g_{\phi_{x,\theta}^t}\left(x,\theta\right)$ denotes the gradient of $\ell_\theta\left(\phi_x\right)$ w.r.t. ϕ. Therefore, after T steps of the iteration, we have the function $z_\theta^T\left(x,\xi\right)$ from $\mathbb{R}^d\times\Xi$ to $\mathbb{R}^r$ as

$$z_\theta^T\left(x,\xi\right) = \mu_{\phi_x^T}\left(\xi\right), \quad \xi\sim p\left(\xi\right), \tag{27}$$

where $\phi_x^0 = h_W(x)$. Plug (27) into $L(\theta)$, we have the surrogate objective $\tilde{L}(\theta, W)$ defined as

$$\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim p(\xi)}\left[\log p_\theta\left(x, \mu_{\phi_x^T}(\xi)\right) + \log p(\xi) - \log\left|\det\frac{\partial\mu_{\phi_x}(\xi)}{\partial\xi}\right|\Bigg|_{\phi_x=\phi_x^T}\right],$$

where $\log\left|\det\frac{\partial\mu_{\phi_x}(\xi)}{\partial\xi}\right|\Bigg|_{\phi_x=\phi_x^T}$ denotes the log-determinant value of $\frac{\partial\mu_{\phi_x}(\xi)}{\partial\xi}$ with ϕ_x set to be ϕ_x^T.

We can follow the chain-rule to calculate the gradient estimator w.r.t. θ and W for $\tilde{L}(\theta, W)$, and apply the SGD to optimize $\tilde{L}(\theta, W)$ w.r.t. both θ and W.

B.4 Parametric Optimization Embedding

We have demonstrated the optimization embedding to generate the nonparametric variational distributions, either in the form of arbitrary flow or individual pre-fixed distribution for each sample. In fact, the optimization embedding is so general that its stochastic variant can be even applied to the parameters of the vanilla amortized VAE. Specifically, we take $q_\phi(z|x) = \mathcal{N}\left(\mu_{\phi_1}(x), \mathrm{diag}\left(\sigma^2_{\phi_2}(x)\right)\right)$ as an example, the ELBO becomes

$$\max_\phi \mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim\mathcal{N}(0,\mathbf{I})}\left[\log p_\theta\left(x|\mu_{\phi_1}(x) + \sigma_{\phi_2}(x)\cdot\xi\right) + \frac{1}{2}\cdot\mathbf{1}^\top\left(2\log\sigma_{\phi_2}(x) - \mu_{\phi_1}(x) - \sigma^2_{\phi_2}(x)\right)\right]. \tag{28}$$

Since the variable ϕ is global, the calculation of the gradient w.r.t. ϕ requires visiting the whole dataset. However, we can use stochastic gradient. Then, the stochastic optimization procedure for global parameter ϕ from ϕ^0 can be embedded as

$$\phi_\theta^{t+1} = f^{-1}\left(\eta_t\widehat{g}_{\phi_\theta^t}(\theta) + f\left(\phi_\theta^t\right)\right), \tag{29}$$

where $\widehat{g}$ denote the stochastic approximation of the true gradient. Plug $\phi^T(\theta)$ into $L\left(\theta, \phi^0\right)$, we have the surrogate objective $\tilde{L}\left(\theta, \phi^0\right)$ defined as

$$\mathbb{E}_{x\sim\mathcal{D}}\mathbb{E}_{\xi\sim\mathcal{N}(0,\mathbf{I})}\left[\log p_\theta\left(x|\mu_{\phi^T_{1,\theta}}(x) + \sigma_{\phi^T_{2,\theta}}(x)\cdot\xi\right) + \frac{1}{2}\cdot\mathbf{1}^\top\left(2\log\sigma_{\phi^T_{2,\theta}}(x) - \mu_{\phi^T_{1,\theta}}(x) - \sigma^2_{\phi^T_{2,\theta}}(x)\right)\right].$$

We can follow the chain-rule to calculate the gradient estimator for $\tilde{L}\left(\theta, \phi^0\right)$, and apply the SGD to optimize w.r.t. θ and ϕ^0.

The significant difference between the parametric CVB and the CVB with Gaussian latent variable in Section B.1 is the optimization embedding objects: the former is w.r.t. the *global* amortized variational distribution parameters, while the latter one is w.r.t. the *local* variables for each sample x. Meanwhile, the key difference between parametric CVB and the vanilla amortized VAE is that the calculation of the gradient w.r.t. θ now need to back-propagate through ϕ_θ^T. With the increasing of T, the computation cost will be increasing. Therefore, in practice, one needs to balance the embedding accuracy and the computational cost by tuning T.

C Hybrid CVB

As we discussed in Section 4, the optimization embedding with optimal dual and the Langevin dynamics are all follows the gradient flow in 2-Wasserstein metric, just with different Fokker-Plank equations. Therefore, it is nature to combine the proposed optimization embedding with Langevin dynamcis, therefore, we have a stochastic mapping from $\mathbb{R}^d \times \Xi \to \mathbb{R}^r$ as

$$z^t_{x,\xi^0;\theta} = (1-\lambda) f^{-1}\left(\eta_t g\left(x, z^{t-1}_{x,\xi^0;\theta}\right) + f\left(z^{t-1}_{x,\xi^0;\theta}\right)\right) \tag{30}$$

$$+ \lambda\left(z^{t-1}_{x,\xi;\theta} + \eta_t\nabla\log p_\theta(x,z) + 2\sqrt{\eta}\xi^{t-1}\right), \tag{31}$$

where $\xi^i \sim \mathcal{N}(0, I)$ and $\lambda \in \{0, 1\}$.

We can replace the optimization embedding in Algorithm 1 with the hybrid embedding, which achieves the hybrid CVB.

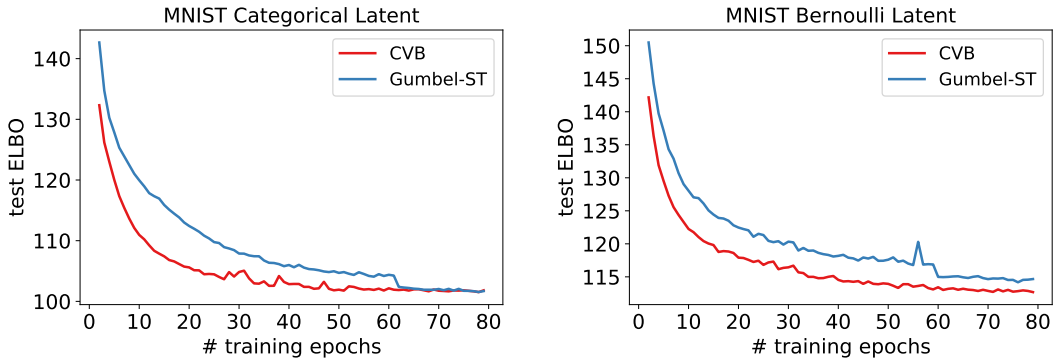


Figure 4: Convergence speed comparison in terms of number epoch on MNIST for discrete latent variable models. The CVB achieves better test ELBO with faster convergence speed comparing to the original Gumbel-Softmax parametrization for both categorial and Bernoulli distributed latent variables.

D Additional Experiments

D.1 Generality on Discrete Latent Variable Models

We test the CVB on categorical and binary latent variable model on MNIST. We utilize the Gumbel-Softmax to relax the distribution over the discrete latent variables and apply the optimization embedding variant introduced in Section B.2. We conduct comparison between the CVB on discrete latent variable models with the VAE with Gumbel-Softmax reparametrization trick [Jang et al., 2016], which is the current state-of-the-art. The results are illustrated in Figure 4. We can see that the parametrized CVB achieves better performance in a faster speed.

D.2 Additional Results on Generative Ability

The additional experimental results on MNIST and CelebA are illustrated in Figure 5 and Figure 6.

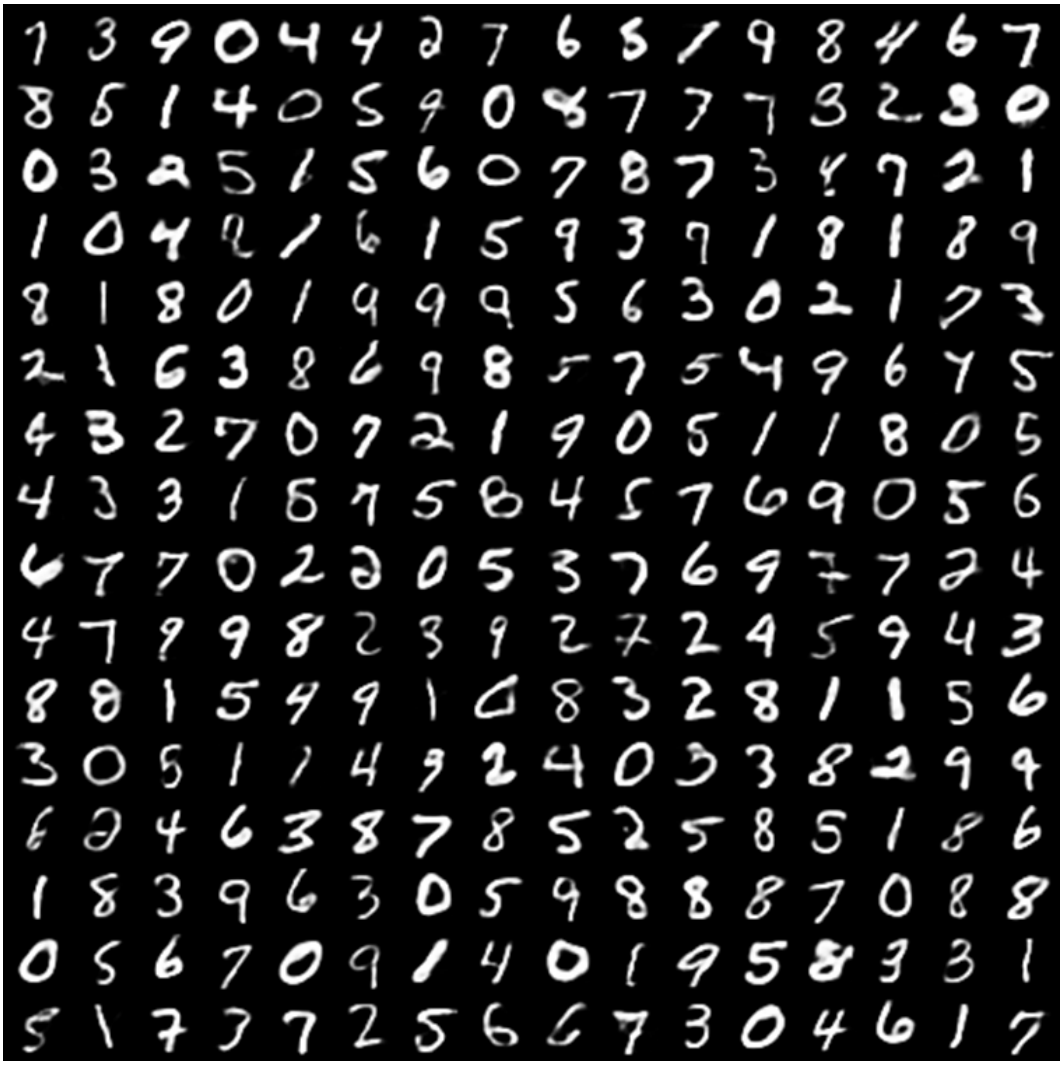


Figure 5: Generated images for MNIST dataset by CVB.

Figure 6: Generated images for CelebA dataset by CVB.