2nd Symposium on Advances in Approximate Bayesian Inference, 2019 1–20

A Wasserstein Minimum Velocity Approach to Learning Unnormalized Models

Ziyu Wang WZY196@GMAIL.COM
Shuyu Cheng CHENGSY18@MAILS.TSINGHUA.EDU.CN
Yueru Li LIYR18@MAILS.TSINGHUA.EDU.CN
Jun Zhu DCSZJ@TSINGHUA.EDU.CN
Bo Zhang DCSZB@TSINGHUA.EDU.CN
Department of Computer Science and Technology
Tsinghua University
Beijing, 100084 China

Abstract

Score matching provides an effective approach to learning flexible unnormalized models, but its scalability is limited by the need to evaluate a second-order derivative. In this paper, we connect a general family of learning objectives including score matching to Wasserstein gradient flows. This connection enables us to design a scalable approximation to these objectives, with a form similar to single-step contrastive divergence. We present applications in training implicit variational and Wasserstein auto-encoders with manifold-valued priors.

1. Introduction

Unnormalized models define the model distribution as $q(x;\theta) \propto \exp(-\mathcal{E}(x;\theta))$, where $\mathcal{E}(x;\theta)$ is an *energy function* that can be parameterized by e.g. DNNs. Unnormalized models can be used directly for density estimation, but another important application is in gradient estimation for implicit variational inference, where we can use score estimation in latent space to approximate an intractable learning objective. This approach leads to improved performance in training implicit auto-encoders (Song et al., 2019).

Maximum likelihood estimation for unnormalized models is intractable, and *score matching* (Hyvärinen, 2005) is a popular alternative. Score matching optimizes the Fisher divergence

$$D_F(p|q) := \frac{1}{2}\mathbb{E}_{p(x)}\left[\|\nabla_x \log p(x) - \nabla_x \log q(x;\theta)\|^2\right], \tag{1}$$

where we denote the data distribution as p. Hyvärinen (2005) shows D_F is equivalent to $\mathbb{E}_{p(x)}\left[\Delta \log q(x;\theta) + \frac{1}{2}\|\nabla \log q(x;\theta)\|^2\right]$, where $\Delta = \sum_i \partial_i^2$ is the Laplacian; the equivalent form can be estimated using samples from p. So far, when $\mathcal{E}$ has a complex parameterization, calculating the equivalent objective is still difficult, as it involves the second-order derivatives; and in practice, people turn to scalable approximations of the score matching objective (Song et al., 2019; Hyvarinen, 2007; Vincent, 2011) or other objectives such as the kernelized Stein discrepancy (KSD; Liu et al., 2016b; Liu and Wang, 2017). However, these approximations are developed on a case-by-case basis, leaving important applications unaddressed; for example, there is a lack of scalable learning methods for models on manifolds (Mardia et al., 2016).

In this work, we present a unifying perspective to this problem, and derive scalable approximations for a variety of objectives including score matching. We start by interpreting these objectives as the initial *velocity* of certain distribution-space gradient flows, which are simulated by common samplers. This novel interpretation leads to a scalable approximation algorithm for all such objectives, reminiscent to single-step contrastive divergence (CD-1).

We refer to any objective bearing the above interpretation as above as a "minimum velocity learning objective", a term coined in the unpublished work Movellan (2007). Our formulation is a distribution-space generalization of their work, and applies to different objectives as the choice of distribution space varies. Another gap we fill in is the development of a practically applicable algorithm: while the idea of approximating score matching with CD-1 is also explored in (Hyvarinen, 2007; Movellan, 2007), previously the approximation suffers from an infinite variance problem, and is thus believed to be impractical (Hyvarinen, 2007; Saremi et al., 2018); we present a simple fix to this issue. Additionally, we present an approximation to the objective function instead of its gradient, thus enabling the use of regularization like early-stopping. Other related work will be reviewed in Appendix C.

One important application of our framework is in learning unnormalized models on manifolds. This is needed in areas such as image analysis (Srivastava et al., 2007), geology (Davis and Sampson, 1986) and bioinformatics (Boomsma et al., 2008). Moreover, as we present an approximation to the Riemannian score matching objective, it enables flexible inference for VAEs and WAEs with manifold-valued latent variables, as it enables gradient estimation for implicit variational distributions on manifolds. It is believed that auto-encoders with a manifold-valued latent space can capture the distribution of certain types of data better (Mathieu et al., 2019; Anonymous, 2020; Davidson et al., 2018). As we will see in Section 3, our method leads to improved performance of VAEs and WAEs.

2. Wasserstein Minimum Velocity Learning

We now present our framework, which concerns all learning objectives of the following form:

$$L_{\text{mvl}}(\theta) := -\left.\frac{d}{dt}\text{KL}(p_t\|q_\theta)\right|_{t=0}, \tag{2}$$

where q_θ is the model distribution, $\{p_t\}$ is the gradient flow of KL_q in a suitable distribution space (e.g. the 2-Wasserstein space), and KL_q is the exclusive KL divergence functional, $p \mapsto \text{KL}(p\|q_\theta)$. We refer to these objectives as "minimum velocity learning (MVL) objectives", since (3) will show that they correspond to the initial velocity of the gradient flow.

(2) subsumes the Fisher divergence for score matching as a special case, since from the properties of the 2-Wasserstein space (*please refer to Appendix A for the necessary preliminary knowledge*), we have $D_F(p|q) = \frac{1}{2}\|\text{grad}_p\text{KL}_q\|^2$, where the gradient and norm are defined in the 2-Wasserstein space, and the data manifold $\mathcal{X}$ is endowed with the Euclidean metric. Rearranging terms, we get

$$\|\text{grad}_p\text{KL}_q\|^2 = d(\text{KL}_q)_p(\text{grad}_p\text{KL}_q) = -\left.\frac{d}{dt}\text{KL}(p_t\|q_\theta)\right|_{t=0} = L_{\text{mvl}}(\theta). \tag{3}$$

i.e., score matching is a special case of the MVL objective, when the space of distributions is chosen as the 2-Wasserstein space $\mathcal{P}(\mathcal{X})$.

2.1. Scalable Approximation to the MVL Objectives

In certain cases, the gradient flow of KL_q corresponds to common samplers, and can be efficiently simulated: e.g., the gradient flow in $\mathcal{P}(\mathcal{X})$ is the (Riemannian) Langevin dynamics. Now we utilize this connection to design a scalable approximation to these objectives.

First, note (3) holds regardless of the chosen space of distributions. Denote $\mathcal{H}[p] := \mathbb{E}_p \log p$, $\mathcal{F}[p] := \mathbb{E}_p \log q = -\mathbb{E}_p \mathcal{E}$, so $\mathrm{KL}_q = \mathcal{H} - \mathcal{F}$, then by linearity we have

$$L_{\mathrm{mvl}}(\theta) = \|\mathrm{grad}_p \mathrm{KL}_q\|^2 = \|\mathrm{grad}_p \mathcal{H}\|^2 - 2\langle \mathrm{grad}_p \mathcal{F}, \mathrm{grad}_p \mathrm{KL}_{q^{1/2}} \rangle. \tag{4}$$

As the first term in (4) is independent of θ, the MVL objective is always equivalent to the second term. We approximate it by simulating a modified gradient flow: let $\tilde{p}_t$ be the distribution obtained by running the sampler targeting $q^{1/2}$. Then

$$\langle \mathrm{grad}_p \mathcal{F}, -\mathrm{grad}_p \mathrm{KL}_{q^{1/2}} \rangle = (d\mathcal{F})_p(-\mathrm{grad}_p \mathrm{KL}_{q^{1/2}}) = \lim_{\epsilon \to 0} \frac{\mathbb{E}_{\tilde{p}_t} \log q_\theta - \mathbb{E}_p \log q_\theta}{\epsilon}. \tag{5}$$

(5) can be approximated by replacing the limit with a fixed ϵ, and running the corresponding sampler starting from a mini-batch of training data. The approximation becomes unbiased when $\epsilon \to 0$.

A Control Variate The approximation to (5) will have small bias. However, it suffers from high variance when the sampler consists of Itô diffusion (e.g. when it is Langevin dynamics). Fortunately, we can solve this problem with a control variate.

To illustrate this, suppose the MVL objective is defined using Langevin dynamics (LD). Without loss of generality, we assume we use a batch size of 1 in the approximation, so approximation is $\hat{L}_\epsilon = \frac{2}{\epsilon}\left[\mathcal{E}(x^+) - \mathcal{E}\left(x^+ - \epsilon \nabla_x \frac{\mathcal{E}(x^+)}{2} + \sqrt{2\epsilon} Z\right)\right]$, where x^+ is sampled from the training data, and $Z \sim \mathcal{N}(0, I)$. By Taylor expansion[1],

$$\frac{1}{2}\hat{L}_\epsilon = \frac{\|\nabla_x \mathcal{E}(x^+)\|^2}{2} - Z^\top \nabla_x^2 \mathcal{E}(x^+) Z - \sqrt{\frac{2}{\epsilon}} Z^\top \nabla_x \mathcal{E}(x^+) + o(1),$$

and as $\epsilon \to 0$, $\mathrm{Var}\, \hat{L}_\epsilon = \Theta\left(\epsilon^{-1}\right) \to \infty$. Thus a control variate is needed. In this LD example, the control variate is $\sqrt{2\epsilon^{-1}} Z^\top \nabla_x \mathcal{E}(x^+)$; More generally, the control variate is always the inner product of $\nabla_x \mathcal{E}(x^+)$ and the diffusion term in the sampler.

As a side product of our work, we note that similar control variate can be obtained for CD-1 and denoising score matching, and it solves their problem. See Appendix B.2.

2.2. Application: Density and Score Estimation on Manifold

As an application, let us consider learning unnormalized models on Riemannian manifolds. In this case, the gradient flow of KL_q in the 2-Wasserstein space becomes the Riemannian Langevin dynamics (Xifara et al., 2014), and the approximate MVL objective becomes

$$L_{\text{mvl-rld}} = \frac{2}{\epsilon}\big(\mathcal{E}(y^-; \theta) - \mathcal{E}(y; \theta) - \underbrace{\sqrt{2\epsilon}\partial_i \mathcal{E}(y) z^i}_{\text{control variate}}\big), \qquad \text{where} \tag{6}$$

$$(y^-)^i = y^i + \epsilon\left(-g^{ij}\partial_j \frac{\mathcal{E}(y;\theta) + \log|G(y)|}{2} + \partial_k g^{ik}\right) + \sqrt{2\epsilon} z^i, \tag{7}$$

1. We need to expand to the second order when the increment is a discretization of some Itô diffusion.

is a sample from the Riemannian LD, and $z \sim \mathcal{N}(0, G^{-1}(y))$. This is the Riemannian score matching objective (Mardia et al., 2016), which also has the form of (1), but the norm is defined by the Riemannian metric of $\mathcal{X}$. From this example, we can see the power of our framework, which enables us to approximate new objectives with ease.

3. Training Auto-encoders with Manifold-valued Prior

We now apply our approximation to learning implicit variational auto-encoders (VAEs) and Wasserstein auto-encoders (WAEs) with manifold-valued prior.

First we review the use of score estimator in learning implicit auto-encoding models. We use VAE as an example; for WAE with KL penalty, the derivation is similar, see Song et al. (2019). The VAE objective is $\mathbb{E}_{p(x)}\mathbb{E}_{q(z|x;\phi)} \log \frac{p(z)p(x|z;\theta)}{q(z|x;\phi)}$, where $q(z|x;\phi)$ is the push-forward measure of $\mathcal{N}(0, I)$ by $f(\cdot;\phi)$. The objective is intractable, as the entropy term $H[q(z|x;\phi)]$ is intractable; however, we can show that (Li and Turner, 2018)

$$\nabla_\phi H[q(z)] = -\mathbb{E}_\epsilon \left[\nabla_z \log q(z) \nabla_\phi f_q(\epsilon;\phi)\right]. \tag{8}$$

Thus to approximate the objective, it suffices to approximate the score function $\nabla_z \log q(z)$. This can be implemented by learning an unnormalized model using score matching. As the score matching objective directly aligns the learnt score function $\nabla_z \mathcal{E}$ to the data score, it can be viewed as score estimation using conservative fields, thus it leads to a better approximation to the gradient $\nabla_\phi H[q(z)]$ compared to indirect approximations such as the adversarial density ratio estimators.

As we turn to the case where the latent space is an embedded manifold, the original score matching objective can no long be used, since $q(z)$ no longer has a density w.r.t. the Lebesgue measure in the embedded space. However, we can still do score estimation on the manifold, i.e. estimate the log derivative of the density w.r.t. the manifold Hausdorff measure. This can be done by fitting an unnormalized model on manifold, using the approximate objective developed in Section 2.2. We note that in this case, (8) still holds, and we can still estimate the gradient of the ELBO using the score estimate. See Appendix E.4 for details.

Empirical Evaluations We apply our method to train implicit hyperspherical VAEs (Davidson et al., 2018) with implicit encoders and WAEs, on the MNIST dataset. Our experiment setup follows Song et al. (2019), with the exception that we parameterize an *energy network* $\mathcal{E}(z;\psi)$ and uses its gradient as the score estimate, instead of parameterizing a score network. Detailed setup and additional synthetic experiments are in Appendix D.

For the VAE experiment, we compare with VAEs with explicit variational posteriors, as well as Euclidean VAEs, and report negative log likelihood estimated with annealed importance sampling (Neal, 2001); for the WAE experiment, we compare with WAE-GAN (Arjovsky et al., 2017), and report the FID score (Heusel et al., 2017). The results are summarized in Figure 1. We can see that in all cases, hyperspherical prior outperforms the Euclidean prior, and our method leads to improved performance. Interestingly, for VAEs with explicit encoders, hyperspherical VAE could not match the performance of Euclidean VAE in high dimensions; this is consistent with the result in Davidson et al. (2018), who incorrectly conjectured that hyperspherical prior is inadequate in high dimensions; we can see that the problem is actually the lack of flexibility in inference, which our method addresses.

VAE	$n_z = 8$		$n_z = 32$	
Method	Euc.	Sph.	Euc.	Sph.
Explicit	96.47	95.38	90.11	91.16
Implicit	95.71	**94.99**	90.17	**88.63**

WAE	$n_z = 8$	
Method	Euc.	Sph.
GAN	25.48	20.40
MVL (Ours)	21.95	**19.13**

Figure 1: Left: negative log likelihood in the VAE experiment. Right: FID score in the WAE experiment. **Boldface** indicates the best result.

References

Anonymous. Poincaré wasserstein autoencoder. In *Submitted to International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=BJgLpaEtDS. under review.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223, 2017.

Alessandro Barp, Francois-Xavier Briol, Andrew B. Duncan, Mark Girolami, and Lester Mackey. Minimum stein discrepancy estimators, 2019.

Wouter Boomsma, Kanti V Mardia, Charles C Taylor, Jesper Ferkinghoff-Borg, Anders Krogh, and Thomas Hamelryck. A generative, probabilistic model of local protein structure. *Proceedings of the National Academy of Sciences*, 105(26):8932–8937, 2008.

Simon Byrne and Mark Girolami. Geodesic monte carlo on embedded manifolds. *Scandinavian Journal of Statistics*, 40(4):825–845, 2013.

Changyou Chen, Ruiyi Zhang, Wenlin Wang, Bai Li, and Liqun Chen. A unified particle-optimization framework for scalable bayesian sampling. *arXiv preprint arXiv:1805.11659*, 2018.

Tim R Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M Tomczak. Hyperspherical variational auto-encoders. *arXiv preprint arXiv:1804.00891*, 2018.

John C Davis and Robert J Sampson. *Statistics and data analysis in geology*, volume 646. Wiley New York et al., 1986.

Herbert Federer. *Geometric measure theory*. Springer, 2014.

Jackson Gorham, Andrew B. Duncan, Sebastian Vollmer, and Lester Mackey. Measuring sample quality with diffusions. *Annals of Applied Probability*, April 2019.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 297–304, 2010.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017.

Elton P Hsu. *Stochastic analysis on manifolds*, volume 38. American Mathematical Soc., 2002.

Elton P Hsu. A brief introduction to brownian motion on a riemannian manifold. *lecture notes*, 2008.

Michael F Hutchinson. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics-Simulation and Computation*, 19(2):433–450, 1990.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(Apr):695–709, 2005.

Aapo Hyvarinen. Connections between score matching, contrastive divergence, and pseudo-likelihood for continuous-valued variables. *IEEE Transactions on neural networks*, 18(5): 1529–1531, 2007.

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

Chang Liu, Jun Zhu, and Yang Song. Stochastic gradient geodesic mcmc methods. In *Advances in Neural Information Processing Systems*, pages 3009–3017, 2016a.

Chang Liu, Jingwei Zhuo, Pengyu Cheng, Ruiyi Zhang, Jun Zhu, and Lawrence Carin. Understanding and accelerating particle-based variational inference. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4082–4092, Long Beach, California USA, 09–15 Jun 2019a. PMLR.

Chang Liu, Jingwei Zhuo, and Jun Zhu. Understanding mcmc dynamics as flows on the wasserstein space. *arXiv preprint arXiv:1902.00282*, 2019b.

Qiang Liu. Stein variational gradient descent as gradient flow. In *Advances in neural information processing systems*, pages 3115–3123, 2017.

Qiang Liu and Dilin Wang. Stein variational gradient descent: A general purpose bayesian inference algorithm. In *Advances in neural information processing systems*, pages 2378–2386, 2016.

Qiang Liu and Dilin Wang. Learning deep energy models: Contrastive divergence vs. amortized mle. *arXiv preprint arXiv:1707.00797*, 2017.

Qiang Liu, Jason Lee, and Michael Jordan. A kernelized stein discrepancy for goodness-of-fit tests. In *International conference on machine learning*, pages 276–284, 2016b.

Yulong Lu, Jianfeng Lu, and James Nolen. Accelerating Langevin Sampling with Birth-death. *arXiv e-prints*, art. arXiv:1905.09863, May 2019.

Yi-An Ma, Tianqi Chen, and Emily Fox. A complete recipe for stochastic gradient mcmc. In *Advances in Neural Information Processing Systems*, pages 2917–2925, 2015.

Kanti V Mardia, John T Kent, and Arnab K Laha. Score matching estimators for directional distributions. *arXiv preprint arXiv:1604.08470*, 2016.

Emile Mathieu, Charline Le Lan, Chris J Maddison, Ryota Tomioka, and Yee Whye Teh. Continuous hierarchical representations with poincaré variational auto-encoders. In *Advances in neural information processing systems*, 2019.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*, 2018.

Javier R Movellan. A minimun velocity approach to learning. *unpublished*, 2007.

Radford M Neal. Annealed importance sampling. *Statistics and computing*, 11(2):125–139, 2001.

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

Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

Martin Raphan and Eero P Simoncelli. Least squares estimation without priors or supervision. *Neural computation*, 23(2):374–420, 2011.

Francisco JR Ruiz and Michalis K Titsias. A contrastive divergence for combining variational inference and mcmc. *arXiv preprint arXiv:1905.04062*, 2019.

Saeed Saremi, Arash Mehrjou, Bernhard Schölkopf, and Aapo Hyvärinen. Deep energy estimator networks. *arXiv preprint arXiv:1805.08306*, 2018.

Jiaxin Shi, Shengyang Sun, and Jun Zhu. A spectral approach to gradient estimation for implicit distributions. In *Proceedings of the 35th International Conference on Machine Learning*, pages 4651–4660, 2018.

Jascha Sohl-Dickstein, Peter Battaglino, and Michael R DeWeese. Minimum probability flow learning. In *Proceedings of the 28th International Conference on International Conference on Machine Learning*, pages 905–912. Omnipress, 2011.

Yang Song, Sahaj Garg, Jiaxin Shi, and Stefano Ermon. Sliced score matching: A scalable approach to density and score estimation. *arXiv preprint arXiv:1905.07088*, 2019.

Bharath Sriperumbudur, Kenji Fukumizu, Arthur Gretton, Aapo Hyvärinen, and Revant Kumar. Density estimation in infinite dimensional exponential families. *The Journal of Machine Learning Research*, 18(1):1830–1888, 2017.

Anuj Srivastava, Ian Jermyn, and Shantanu Joshi. Riemannian analysis of probability density functions with applications in vision. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2007.

Amirhossein Taghvaei and Prashant Mehta. Accelerated flow for probability distributions. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6076–6085, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

C. Villani. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2008. ISBN 9783540710509. URL https://books.google.co.jp/books?id=hV8o5R7_5tkC.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

Li Wenliang, Dougal Sutherland, Heiko Strathmann, and Arthur Gretton. Learning deep kernels for exponential family densities. *arXiv preprint arXiv:1811.08357*, 2018.

Tatiana Xifara, Chris Sherlock, Samuel Livingstone, Simon Byrne, and Mark Girolami. Langevin diffusions and the metropolis-adjusted langevin algorithm. *Statistics & Probability Letters*, 91:14–19, 2014.

Jianyi Zhang, Ruiyi Zhang, and Changyou Chen. Stochastic particle-optimization sampling and the non-asymptotic convergence theory. *arXiv preprint arXiv:1809.01293*, 2018.

Appendix A. Preliminary Knowledge

In this section, we review background knowledge needed in this work, most importantly Wasserstein gradient flow and its connection to sampling algorithms.

A.1. Manifolds, Flows and the 2-Wasserstein Space

A (differential) *manifold* $\mathcal{M}$ is a topological space locally diffeomorphic to an Euclidean or Hilbert space. A manifold is covered by a set of *charts*, which enables the use of coordinates locally, and specifies a set of basis $\{\partial_i\}$ in the local tangent space. A *Riemannian manifold* further possesses a *Riemannian structure*, which assigns to each tangent space $\mathcal{T}_p\mathcal{M}$ an inner product structure. The Riemannian structure can be described using coordinates w.r.t. local charts.

The manifold structure enables us to differentiate a function along curves. Specifically, consider a curve $c : [0, T] \mapsto \mathcal{M}$, and a smooth function $f : \mathcal{M} \mapsto \mathbb{R}$. At $c(t) \in \mathcal{M}$, a *tangent vector* $\frac{dc}{dt}\big|_t \in \mathcal{T}_{c(t)}\mathcal{M}$ describes the velocity of c passing $c(t)$; the *differential* of the function f at $c(t)$, denoted as $(df)_{c(t)}$, is a *linear* map from $\mathcal{T}_{c(t)}\mathcal{M}$ to $\mathbb{R}$, such that for all c

$$(df)_{c(t_0)}\left(\frac{dc}{dt}\bigg|_{t_0}\right) = \frac{d}{dt} f(c(t))\bigg|_{t_0}.$$

A *tangent vector field* assigns to each $p \in \mathcal{M}$ a tangent vector $V_p \in \mathcal{T}_p\mathcal{M}$. It determines a *flow*, a set of curves $\{\phi_p(t) : p \in \mathcal{M}\}$ which all have $V_{\phi_p(t)}$ as their velocity. On Riemannian manifolds, the *gradient* of a smooth function f is a tangent vector field $p \mapsto \mathrm{grad}_p f$ such that $\langle \mathrm{grad}_p f, v\rangle = (df)_p(v)$ for all $v \in T_p\mathcal{M}$. It determines the *gradient flow*, which generalizes the Euclidean-space notion $dx = \nabla_x f(x)dt$.

We will work with two types of manifolds: the data space $\mathcal{X}$ when we apply our method to manifold-valued data, and the space of probability distributions over $\mathcal{X}$. On the space of distributions, we are mostly interested in the *2-Wasserstein space* $\mathcal{P}(\mathcal{X})$, a Riemannian manifold. The following properties of $\mathcal{P}(\mathcal{X})$ will be useful for our purposes (Villani, 2008):

1. Its tangent space $\mathcal{T}_p\mathcal{P}(\mathcal{X})$ can be identified as a subspace of the space of vector fields on $\mathcal{X}$; the Riemannian metric of $\mathcal{P}(\mathcal{X})$ is defined as

$$\langle X, Y\rangle_p := \mathbb{E}_{p(u)}\langle X(u), Y(u)\rangle_u, \tag{9}$$

   for all $p \in \mathcal{P}(\mathcal{X}), X, Y \in \mathcal{T}_p\mathcal{P}(\mathcal{X})$; the inner product on the right hand side above is determined by the Riemannian structure of $\mathcal{X}$.

2. The gradient of the KL divergence functional $\mathrm{KL}_p(q) := \mathrm{KL}(q\|p)$ in $\mathcal{P}(\mathcal{X})$ is

$$(\mathrm{grad}_q \mathrm{KL}_p)(u) = \mathrm{grad}_u \log \frac{q(u)}{p(u)}. \tag{10}$$

We will also consider a few other spaces of distributions, including the Wasserstein-Fisher-Rao space (Lu et al., 2019), and the $\mathcal{H}$-Wasserstein space introduced in (Liu, 2017).

On the data space, we need to introduce the notion of density, i.e. the Radon–Nikodym derivative w.r.t. a suitable base measure. The Hausdorff measure is one such choice; it reduces to the Lebesgue measure when $\mathcal{X} = \mathbb{R}^n$. In most cases, distributions on manifolds

are specified using their density w.r.t. the Hausdorff measure; e.g. "uniform" distributions has constant densities in this sense.

Finally, the data space $\mathcal{X}$ will be embedded in $\mathbb{R}^n$; we refer to real-valued functions on the space of distributions as functionals; we denote the functional $q \mapsto \mathrm{KL}(q\|p)$ as KL_p; we adopt the Einstein summation convention, and omit the summation symbol when an index appears both as subscript and superscript on one side of an equation, e.g. $v^i\partial_i := \sum_i v^i\partial_i$.

A.2. Posterior Sampling by Simulation of Gradient Flows

Now we review the sampling algorithms considered in this work. They include diffusion-based MCMC, particle-based variational inference, and other stochastic interacting particle systems.

Riemannian Langevin Dynamics Suppose our target distribution has density $p(x)$ w.r.t. the Hausdorff measure of $\mathcal{X}$. In a local chart $U \subset \mathcal{X}$, let $G : U \to \mathbb{R}^{m\times m}$ be the coordinate matrix of its Riemannian metric. Then the *Riemannian Langevin dynamics* corresponds to the following stochastic differential equation in the chart[2]:

$$dx = V(x)dt + \sqrt{2G^{-1}(x)}dB_t \tag{11}$$

where

$$V^i(x) = g^{ij}\partial_j\left(\log p(x) - \frac{\log|G(x)|}{2}\right) + \partial_j g^{ij}, \tag{12}$$

and (g^{ij}) is the coordinate of the matrix G^{-1}. It is known (Villani, 2008) that the Riemannian Langevin dynamics is the gradient flow of the KL functional $\mathrm{KL}_p(q) := \mathrm{KL}(q\|p)$ in the 2-Wasserstein space $\mathcal{P}(\mathcal{X})$.

Particle-based Samplers A range of samplers approximate the gradient flow of KL_p in various spaces, using deterministic or stochastic interacting particle systems.[3] For instance, Stein variational gradient descent (SVGD; Liu and Wang, 2016) simulates the gradient flow in the so-called $\mathcal{H}$-Wasserstein space (Liu, 2017), which replaces the Riemannian structure in $\mathcal{P}(\mathcal{X})$ with the RKHS inner product. Birth-death accelerated Langevin dynamics (Lu et al., 2019) is a stochastic interacting particle system that simulates to the gradient flow of KL_p in the Wasserstein-Fisher-Rao space. Finally, the stochastic particle-optimization sampler (SPOS; Zhang et al., 2018; Chen et al., 2018) combines the dynamics of SVGD and Langevin dynamics; as we will show in Appendix E.2, SPOS also has a gradient flow structure.

Appendix B. Wasserstein Minimum Velocity Learning

In this section we present additional discussions about our framework. In Section B.1 we discuss other objectives that can be derived from our framework; in Section B.2 we show

2. (11) differs from the form in some literature (e.g. Ma et al., 2015), as in our case, the density of the target measure is defined w.r.t. the Hausdorff measure of $\mathcal{X}$, instead of the Lebesgue measure in Ma et al. (2015). See (Xifara et al., 2014, eq (12)) or (Hsu, 2008, Section 1.5).
3. There are other particle-based samplers (Liu et al., 2019b,a; Taghvaei and Mehta, 2019) corresponding to accelerated gradient flows. However, as we will be interested in the initial velocity of the flow, they do not lead to new objectives in our framework.

our control variate could be applied to CD-1 and denoising score matching; finally, while readers familiar with Riemannian Brownian motion may be concerned about the use of local coordinates in our Riemannian score matching approximation (6), we show in Section B.3 it does not lead to issues.

B.1. Other MVL Objectives

As our derivation is independent of the distribution space of choice, we can derive approximations to other learning objectives using samplers other than Langevin dynamics, as reviewed in Section A.2. An example is Riemannian Langevin dynamics which we have discussed in the main text; another example is when we choose the sampler as SVGD. In this case, we will obtain an approximation to the kernelized Stein discrepancy, generalizing the derivation in (Liu and Wang, 2017). When the sampling algorithm is chosen as SPOS, the corresponding MVL objective will be an interpolation between KSD and the Fisher divergence. See Appendix E.3 for derivations. Finally, the use of birth-death accelerated Langevin dynamics leads to a novel learning objective.

In terms of applications, our work focuses on learning neural energy-based models, and we find these objectives do not improve over score matching in this aspect. However, these derivations generalize previous discussions, and establish new connections between sampling algorithms and learning objectives. It is also possible that these objectives could be useful in other scenarios, such as learning kernel exponential family models (Sriperumbudur et al., 2017), direct estimation of the score function (Li and Turner, 2018; Shi et al., 2018), and improving the training of GANs (Liu and Wang, 2017) or amortized variational inference methods (Ruiz and Titsias, 2019).

B.2. On CD-1 and Denoising Score Matching: Pitfalls and a Fix

As a side product of our work, we show in this section that our variance analysis explains the pitfall of two well-known approximations to the score matching objective: CD-1 (Hyvarinen, 2007) and denoising score matching (Vincent, 2011). Both approximations become unbiased as a step-size hyper-parameter $\epsilon \to 0$, but could not match the performance of exact score matching in practice, as witnessed in Hyvarinen (2007); Saremi et al. (2018); Song et al. (2019). Our analysis leads to novel control variates for these approximators. As we will show in Section D.1, the variance-reduced versions of the approximations have comparable performance to the exact score matching objective.

Denoising Score Matching (DSM) DSM considers the objective

$$L_{\text{dsm}}(\theta) = \mathbb{E}_{p(x)\mathcal{N}(z|0,I)}\|x + \sigma z - (x + \psi_\theta(x + \sigma z))\|^2. \tag{13}$$

The first two terms inside the norm represent a noise corrupted sample, and ψ_θ represents a "single-step denoising direction" (Raphan and Simoncelli, 2011). It is proved that the optimal ψ satisfies $\psi = \sigma^2 \nabla \log \tilde{p}$, where $\tilde{p}$ is the density of the corrupted distribution (Raphan and Simoncelli, 2011; Vincent, 2011).

Consider the stochastic estimator of (13). We assume a batch size of 1, and denote the data sample as x. To keep notations consistent, denote $\epsilon = \sigma^2$, $\psi_\theta(x) = \epsilon \nabla_x \mathcal{E}(x; \theta)$. Then

the stochastic estimator is

$$\hat{L}_{\text{dsm}} = \|x + \sqrt{\epsilon}z - \epsilon\nabla_x \mathcal{E}(x + \sqrt{\epsilon}z; \theta) - x\|^2.$$

Denote $\tilde{x} := x + \sqrt{\epsilon}z$. By Taylor expansion we have

$$\begin{aligned}
\hat{L}_{\text{dsm}} &= \|x + \sqrt{\epsilon}z - \epsilon\nabla\mathcal{E}(x + \sqrt{\epsilon}z) - x\|^2 && (14)\\
&= \epsilon\|z\|^2 + \epsilon^2\|\nabla\mathcal{E}(\tilde{x})\|^2 - 2\epsilon^{3/2}\langle z, \nabla\mathcal{E}(\tilde{x})\rangle && (15)\\
&= \epsilon\|z\|^2 + \epsilon^2\|\nabla\mathcal{E}(\tilde{x})\|^2 - 2\epsilon^{3/2}\langle z, \nabla\mathcal{E}(x) + (\nabla^2\mathcal{E}(x))(\sqrt{\epsilon}z) + O(\epsilon)\rangle && (16)\\
&= \underbrace{\epsilon^2\left(\|\nabla\mathcal{E}(\tilde{x})\|^2 - 2z^\top(\nabla^2\mathcal{E}(x))z\right)}_{A} + \underbrace{\epsilon\|z\|^2 - 2\epsilon^{3/2}z^\top\nabla\mathcal{E}(x)}_{B} + o(\epsilon^2), && (17)
\end{aligned}$$

As

$$\mathbb{E}_z(z^\top\nabla^2\mathcal{E}(x)z) = \Delta\mathcal{E}(x)$$

which is known as the Hutchinson's trick (Hutchinson, 1990),

$$\lim_{\epsilon\to 0}\epsilon^{-2}\mathbb{E}(A) = 2D_F(p|q).$$

But $Var(B) = O(\epsilon^2)$, so as $\epsilon \to 0$, the rescaled estimator $\epsilon^{-2}\hat{L}_{\text{dsm}}$ becomes unbiased with *infinite variance*; and subtracting (B) from (A) results in a finite-variance estimator.

CD-1 with Langevin Dynamics Proposed as an approximation to the maximum likelihood estimate, the K-step contrastive divergence (CD-K) learning rule updates the model parameter with

$$\theta_{\ell+1} \leftarrow \theta_\ell + \nu\left[\mathbb{E}_p\partial_\theta\mathcal{E} - \mathbb{E}_{p_K}\partial_\theta\mathcal{E}\right], \tag{18}$$

where ν is the learning rate, and p_K is obtained from p by running K steps of MCMC. (18) does not define a valid objective, since p_K also depends on θ; however, Hyvarinen (2007) proved that when $K = 1$ and the sampler is the Langevin dynamics, (18) recovers the *gradient* of the score matching objective.

Using the same derivation as in the main text, we can see that as the step-size of the sampler approaches 0 (and ν is re-scaled appropriately), the gradient produced by CD-1 also suffers from infinite variance, and this can be fixed using the same control variate.

However, practical utility of CD-1 is still hindered by the fact that it does not correspond to a valid learning *objective*; consequently, it is impossible to monitor the training process for CD-1, or introduce regularizations such as early stopping[4].

B.3. Justification of the Use of Local Coordinates in (6)

Readers familiar with Riemannian Brownian motion will notice that we used local coordinates when deriving the MPF objective, and this is only valid until the particle exits the local chart. In this section, we show that this does not affect the validity of our method; specifically, we prove in Proposition 3 that the local coordinate representation lead to valid approximation to the MVL objective in the compact case. We also argue in Remark 4 that the use of local coordinate does not lead to numerical instability.

4. In practice, the term $\mathbb{E}_p\mathcal{E} - \mathbb{E}_{p_K}\mathcal{E}$ is often used to tract the training process of CD-K. It is not a proper loss; we can see from (4) that when $K = 1$ and $\epsilon \to 0$, $\mathbb{E}_p\mathcal{E} - \mathbb{E}_{p_K}\mathcal{E}$ is significantly different from the proper score matching (MVL) loss, by a term of $\frac{1}{2}\|\text{grad}_p\mathcal{F}\|^2$.

Remark 1 *While a result more general than Proposition 3 is likely attainable (e.g. by replacing compactness of $\mathcal{X}$ with quadratic growth of the energy), this is out of the scope of our work; for our purpose, it is sufficient to note that the proposition covers manifolds like S^n, and the local coordinate issue will not exist in manifolds possessing a global chart, such as H^n.*

Lemma 2 ***(Theorem 3.6.1 in (Hsu, 2002))*** *For any manifold $\mathcal{M}$, $x \in \mathcal{M}$, and a normal neighborhood B of x, there exists constant $C > 0$ such that the first exit time τ from B, of the Riemannian Brownian motion starting from x, satisfies*

$$P\left(\tau \leq \frac{C}{L}\right) \leq e^{-L/2}$$

for any $L \geq 1$.

Proposition 3 *Assume the data manifold $\mathcal{X}$ is compact, and for all θ, $\mathcal{E}(\cdot;\theta)$ is in C^1. Let $\tilde{L}_{\text{mvl\_rld}}$ be defined as in (6), X_t following the true Riemannian Langevin dynamics targeting $q^{1/2}$. Then*

$$\frac{1}{2}\lim_{\epsilon\to 0}\mathbb{E}(\tilde{L}_{\text{mvl\_rld}}) = \frac{d}{dt}\mathbb{E}(\mathcal{E}(X_t))\Big|_{t=0},$$

i.e. (6) recovers true WMVL objective.

Proof By the tower property of conditional expectation, it suffices to prove the result when $P(X_0 = x) = 1$ for some x. Choose a normal neighborhood B centered at x such that B is contained by our current chart, and has distance from the boundary of the chart bounded by some $\delta > 0$. Let $C, \bar{\tau}$ be defined as in Lemma 2. Recall the Riemannian LD is the sum of a drift and the Riemannian BM. Since $\mathcal{X}$ is compact and $\mathcal{E}$ is in C^1, the drift term in the SDE will have norm bounded by some finite C. Thus the first exit time of the Riemannian LD is greater than $\min(\bar{\tau}, \delta/C) =: \tau$.

Let X_t follow the true Riemannian LD, $\bar{X}_t = X_t$ when $t < \tau$, and be such that $\mathcal{E}(\bar{X}_t) = 0$ afterwards.[5] By Hsu (2008), until τ, $\bar{X}_t$ follows the local coordinate representation of Riemannian LD (11), thus on the event $\{\epsilon \leq \tau\}$, $\bar{X}_\epsilon$ would correspond to y^- in (7). As $\mathcal{X}$ is compact, the continuous energy function $\mathcal{E}$ is bounded by $|\mathcal{E}(\cdot)| \leq A$ for some finite A. Then for sufficiently small ϵ,

$$\begin{aligned}\frac{1}{2}\mathbb{E}(\tilde{L}_{\text{mvl\_rld}}) = \frac{\mathbb{E}(\mathcal{E}(\bar{X}_\epsilon) - \mathcal{E}(X_0))}{\epsilon} &= \frac{\mathbb{E}(\mathcal{E}(X_\epsilon) - \mathcal{E}(X_0))}{\epsilon} + \frac{\mathbb{E}(\mathcal{E}(\bar{X}_\epsilon) - \mathcal{E}(X_\epsilon))}{\epsilon} \\ &= \frac{\mathbb{E}(\mathcal{E}(X_\epsilon) - \mathcal{E}(X_0))}{\epsilon} + \frac{\mathbb{E}(-\mathcal{E}(X_\epsilon)\mathbf{1}_{\{\tau\leq\epsilon\}})}{\epsilon}.\end{aligned}$$

In the above the first term converges to $\frac{d}{dt}\mathbb{E}(\mathcal{E}(X_t))\big|_{t=0}$ as $\epsilon \to 0$, and $\left|\frac{\mathbb{E}(-\mathcal{E}(X_\epsilon)\mathbf{1}_{\{\tau\leq\epsilon\}})}{\epsilon}\right| \leq \frac{A\mathbb{P}(\tau\leq\epsilon)}{\epsilon} = \frac{A\mathbb{P}(\bar{\tau}\leq\epsilon)}{\epsilon} \leq \frac{Ae^{-C/2\epsilon}}{\epsilon} \to 0$ when $\epsilon \to 0$. Hence the proof is complete. ■

5. This is conceptually similar to the standard augmentation used in stochastic process texts; from a algorithmic perspective it can be implemented by modifying the algorithm so that in the very unlikely event when y^- escapes the chart, we return 0 as the corresponding energy. We note that this is unnecessary for manifolds like S^n, since the charts can be extended to $\mathbb{R}^d$ and hence $\tau = \infty$.

Remark 4 *It is argued that simulating diffusion-based MCMC in local coordinates leads to numeric instabilities (Byrne and Girolami, 2013; Liu et al., 2016a). We stress that in our setting of approximating MVL objectives, this is not the case. The reason is that we only need to do a single step of MCMC, with arbitrarily small step-size. Therefore, we could use different step-size for each sample, based on the magnitude of g and $\log q$ in their locations. We can also choose different local charts for each sample, which is justified by the proposition above.*

Appendix C. Related Work

Our work concerns scalable learning algorithms for unnormalized models. This is a longstanding problem in literature, and some of the previous work is discussed in Section 1. Apart from those work, it is worth mentioning Liu and Wang (2017), which also designed a CD-like algorithm to approximate the kernelized Stein discrepancy; as we have discussed in Section B.1, in our framework there exists a similar algorithm, as well as a slight generalization when we replace SVGD with SPOS. Other notable work includes noise contrastive estimation (Gutmann and Hyvärinen, 2010) and Parzen score matching (Raphan and Simoncelli, 2011). However, to our knowledge, they have not been applied to complex unnormalized models such as those parameterized by DNNs, and a comparison would fall out of the scope of this work.

Apart from the MVL formulation used in this work, there also exists other work on the connection between learning objectives of unnormalized model and infinitesimal actions of sampling dynamics. The minimum probability flow framework (Sohl-Dickstein et al., 2011) studies the slightly different objective $\lim_{\epsilon\to 0}\frac{1}{\epsilon}\mathrm{KL}(p_0\|p_\epsilon)$, where $\{p_t\}$ is the trajectory of the sampler. It also recovers score matching as a special instance; however, it does not lead to scalable learning algorithms for *continuous-state* unnormalized models as our method does; instead, its main application is in discrete-state models. Many of the MVL objectives we have derived are also instances of the Stein discrepancy (Gorham et al., 2019; Barp et al., 2019). This interpretation is helpful in establishing theoretical properties, but it does not lead to scalable implementations of these objectives, that do not depend on higher-order derivatives.

Appendix D. Experiment Details and Additional Results

D.1. Synthetic Experiments

To demonstrate the proposed estimators have small bias and variance, we first evaluate them on low-dimensional synthetic data. We also verify the claim in Section B.2 that our control variate improves the performance of CD-1 and DSM.

D.1.1. Approximations to Score Matching

In this section, we evaluate our MVL approximation to the Euclidean score matching objective, as well as the variance-reduced DSM objective[6].

6. An experiment confirming the effectiveness of our control variate on CD-1 is presented in Appendix D.1.3.

Experiment Setup We evaluate the bias and variance of our estimators by comparing them to sliced score matching (SSM), an unbiased estimator for the score matching objective. We choose the data distribution p as the 2-D banana dataset from Wenliang et al. (2018), and the model distribution q_θ as an EBM trained on that dataset. We estimate the squared bias with a stochastic upper bound, using 5,000,000 samples. More specifically, denote the two methods as $\mathbb{E}_{p(x)\mathcal{N}(\epsilon|0,1)}[L_F^{\text{ssm}}(x;\epsilon)]$ and $\mathbb{E}_{p(x)\mathcal{N}(\epsilon|0,1)}[L_F^{\text{mvl}}(x;\epsilon)]$, respectively. We estimate the squared bias as $\frac{1}{K}\sum_{k=1}^{K}\left(\frac{1}{M}\sum_{j=1}^{M}(L_F^{\text{ssm}}(x^{(k)};\epsilon^{(j)}) - L_F^{\text{mvl}}(x^{(k)};\epsilon^{(j)}))\right)^2$, where $x(k) \sim p(x), \epsilon^{(j)} \sim \mathcal{N}(0,1)$ are i.i.d. draws. Observe that this expectation of the estimate upper bounds the true squared bias following Cauchy's inequality; and the bias $\to 0$ as $K, M \to 0$. We choose $K = 100, M = 50000$ and plot the confidence interval. We also use these samples to estimate the variance of our estimator.

For the model distribution q, we choose an EBM as stated in the main text. The energy of the model is parameterized as follows: we parameterize a d-dimensional vector $\psi(x;\theta)$ using a feed-forward network, then return $x^\top\psi(x;\theta)$ as the energy function. This is inspired by the "score network" parameterization in (Song et al., 2019); we note that this choice has little influence on the synthetic experiments (and is merely chosen here for consistency), but leads to improved performance in the AE experiments. Finally, $\psi(x;\theta)$ is parameterized with 2 hidden layers and Swish activation (Ramachandran et al., 2017), and each layer has 100 units. We apply spectral normalization (Miyato et al., 2018) to the intermediate layers. We train the EBM for 400 iterations with our approximation to the score matching objective, using a batch size of 200 and a learning rate of 4×10^{-3}. The choice of training objective is arbitrary; changing it to sliced score matching does not lead to any notable difference, as is expected from this experiment.

Results The results are shown in Figure 2, in which we plot the (squared) bias and variance for both estimators, with varying step-size. The bias plot is shown in the left. We can see that for both estimators, the bias is negligible at $\epsilon \leq 10^{-2}$. We further use a z-test to compare the mean of the two estimators (for $\epsilon = 6 \times 10^{-5}$) with the mean of SSM. The p value is 0.48 for our estimator and 0.19 for DSM, indicating there is no significant difference in either case.

The variance of the estimators, with and without our control variate, are shown in Figure 2 right. As expected, the variance grows unbounded in absence of the control variate, and is approximately constant when it is added. From the scale of the variance, we can see that it is exactly this variance problem that causes the failure of the original DSM estimator.

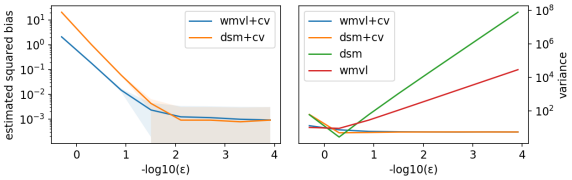

Figure 2: Estimated squared bias and variance of the score matching objective approximations.

D.1.2. Density Estimation on Manifolds

We now evaluate our approximation to the Riemannian score matching objective, by learning an unnormalized model.

Experiment Setup The data distribution is chosen as a mixture of two von Mises distributions on S^1:

$$p(x) = 0.7p_{vM}(x|(0,1),2) + 0.3p_{vM}(x|(0.5,-0.5),3),$$

where p_{vM} is the von Mises density

$$p_{vM}(x|\mu,\sigma) \propto e^{\frac{1}{\sigma^2}\cos(x-\mu)}.$$

The energy function in the model is parameterized with a feed-forward network, using the same score-network-inspired parameterization as in the last experiment. The network uses tanh activation and has 2 hidden layers, each layer with 100 units.

We generate 50,000 samples from $p(x)$ for training. We use full batch training and train for 6,000 iterations, using a learning rate of 5×10^{-4}. The step-size hyperparameter in the MVL approximation is set to 10^{-5}.

Results We plot the log densities of the ground truth distribution as well as the learned model in Figure 3. We can see the two functions matches closely, suggesting our method is suitable for density estimation on manifolds.

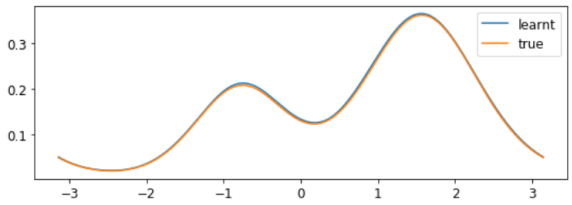

Figure 3: Density estimation on S^1: learned energy vs ground truth in polar coordinates.

D.1.3. On the Variance Problem in CD-1

To verify our control variate also solves the variance issue in CD-1, we train EBMs using CD-1 with varying step-size, with and without our control variate, and compare the score matching loss to EBMs trained with our method as well as sliced score matching. We use a separate experiment for CD-1 since it only estimates the gradient of the score matching loss.

The score matching loss is calculated using SSM on training set, and averaged over 3 separate runs. We use the cosine dataset in (Wenliang et al., 2018); the energy parameterization is the same as in Section D.1.1. The results are shown in Figure 4. We can see that with the introduction of the control variate, CD-1 performs as well as other score matching methods.

D.2. Detailed Setup of the Auto-Encoder Experiments

In all auto-encoder experiments, setup follows from (Song et al., 2019) whenever they applies. The only difference is that for score estimation, we parameterize the energy function, and use

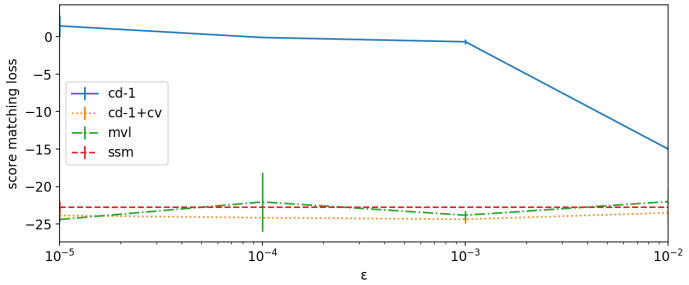

Figure 4: Score matching loss for different methods, with varying step-size. Lower is better.

its gradient as the score estimate, as opposed to directly parameterizing the score function as done in (Song et al., 2019). This modification makes our method applicable; essentially, it corrects the score estimation in (Song et al., 2019) so that it constitute a conservative field, which is a desirable property since score functions should be conservative.

For this reason, we re-implement all experiments for Euclidean-prior auto-encoders to ensure a fair comparison. The results are slightly worse than (Song et al., 2019) for the VAE experiment, but significantly better for WAE experiments. It should be noted that for the VAE experiment, our implicit hyperspherical VAE result is still better than the implicit Euclidean VAE result reported in (Song et al., 2019).

VAE Experiment The (conditional) energy function in this experiment is parameterized using the score-net-inspired method described in Appendix D.1, with a feed-forward network. The network has 2 hidden layers, each with 256 hidden units. We use tanh activation for the network, and do not apply spectral normalization. When training the energy network, we add a L2 regularization term for the energy scale, with coefficient 10^{-4}. The coefficient is determined by grid search on $\{10^{-3}, 10^{-4}, 10^{-5}\}$, using AIS-estimated likelihood on a heldout set created from the training set. The step-size of the MVL approximation is set to 10^{-3}; we note that the performance is relatively insensitive w.r.t. the step-size inside the range of $[10^{-4}, 10^{-2}]$, as suggested by the synthetic experiment. Outside this range, using a smaller step-size makes the result worse, presumably due to floating point errors.

For implicit models, the test likelihood is computed with annealed importance sampling, using 1,000 intermediate distributions, following (Song et al., 2019). The transition operator in AIS is HMC for Euclidean-space latents, and Riemannian LD for hyperspherical latents.

The training setup follows from (Song et al., 2019): for all methods, we train for 100,000 iterations using RMSProp use a batch size of 128, and a learning rate of 10^{-3}.

WAE Experiment on MNIST For our method, the energy network is parameterized in the same way as in the VAE experiments. When training the energy network, we use a step-size of 10^{-4}, and apply L2 regularization on the energy scale with coefficient 10^{-5}.

For the WAE-GAN baseline, we use the Wasserstein GAN (Arjovsky et al., 2017), and parameterize its critic as a feed-forward network with 2 hidden layers, each with 256 units. We experimented with both the standard parameterization (i.e. put a linear layer after the last hidden layer that outputs scalar) as well as the score-network-like parameterization used in our energy network, and found the results to be similar. We use tanh activation, apply spectral normalization and a L2 regularization with coefficient 10^{-4}.

The rest of the training setup follows from (Song et al., 2019): training for 100,000 iterations using RMSProp, a batch size of 128, and a learning rate of 10^{-3}. The Lagrange multiplier hyperparameter λ in the WAE objective is fixed at 10.

WAE Experiment on CelebA The energy network is parameterized in the same way as in (Song et al., 2019). For our method, we use a step-size of 10^{-4}. For the GAN baseline, we use the standard parameterization for the critic, i.e. the final linear layer outputs a scalar; the previous layers follow the same architecture of ours. In both methods we use a L2 regularization with coefficient 10^{-5}. Following (Song et al., 2019), we train for 100,000 iterations, using RMSProp and a learning rate of 10^{-4}. FID scores are calculated using the implementation in (Heusel et al., 2017).

Appendix E. Supporting Results

E.1. On SPOS and MVL

Notations In this section, let the parameter space be d-dimensional, and define $L_2(\rho\mathcal{X} \to \mathbb{R}^d)$ as the space of d-dimensional functions $\{f : \mathbb{E}_{\rho(x)}\|f(x)\|^2 < \infty\}$.

While in the main text, we identified the tangent space of $\mathcal{P}(\mathcal{X})$ as a subspace of $L_2(\rho\mathcal{X} \to \mathbb{R}^d)$ for clarity, here we use the equivalent definition $\mathcal{T}_\rho(\mathcal{P}(\mathcal{X})) := \{s \in L_2(\rho\mathcal{X} \to \mathbb{R}) : \mathbb{E}_\rho s = 0\}$ following (Otto, 2001). The two definition are connected by the transform $s = -\nabla \cdot (\rho p)$ for $p \in L_2(\rho\mathcal{X} \to \mathbb{R}^d)$. Using the new definition, the differential of the KL divergence functional is then $(d\mathrm{KL}_\phi)_\rho(s) := \int s(x) \log \frac{\rho(x)}{\phi(x)} dx$.

E.2. SPOS as Gradient Flow

In this section, we give a formal derivation of SPOS as the gradient flow of the KL divergence functional, with respect to a new metric.

Recall the SPOS sampler targeting distribution (with density) ϕ corresponds to the following density evolution:

$$\partial_t \rho_t = -\nabla \cdot (\rho_t(x) \underbrace{(\phi^*_{\rho_t,\phi}(x) + \alpha \nabla \log(\phi/\rho))}_{\nu_t(x)})$$

where $\alpha > 0$ is a hyperparameter, and

$$\phi^*_{\rho_t,\phi}(x) := \mathbb{E}_{\rho_t(x')}(S_\phi \otimes k)(x', x) := \mathbb{E}_{\rho_t(x')}[(\nabla_{x'} \log \phi(x'))k(x', x) + \nabla_{x'} k(x', x)]$$

is the SVGD update direction (Liu and Wang, 2016; Liu, 2017). Fix ρ, define the integral operator

$$K_\rho[f](x) := \mathbb{E}_{\rho(x')} k(x', x) f(x),$$

and define the tensor product operator $K_\rho^{\otimes d} : L^2(\mathcal{X} \to \mathbb{R}^d) \to L^2(\mathcal{X} \to \mathbb{R}^d)$ accordingly. Then the SVGD update direction satisfies

$$\phi^*_{\rho,\phi} = K_\rho^{\otimes d}[\nabla \log(\phi/\rho)], \tag{19}$$

which we will derive shortly at the end of this subsection. Subsequently, we have

$$\nu_t(x) = (\alpha \mathrm{Id} + K_\rho^{\otimes d})[\nabla \log(\phi/\rho)]. \tag{20}$$

The rest of our derivation follows (Otto, 2001; Liu, 2017): consider the function space $\mathcal{H}_{\rho,\alpha} := \{(\alpha \mathrm{Id} + K_{\rho_t}^{\otimes d})[\nabla h]\}$, where $h : \mathcal{X} \to \mathbb{R}$ is any square integrable and differentiable function. It connects to the tangent space of $\mathcal{P}(\mathcal{X})$ if we consider $s = -\nabla \cdot (\rho \tilde{p})$ for any $\tilde{p} \in \mathcal{H}_{\rho,\alpha}$. Define on $\mathcal{H}_{\rho,\alpha}$ the inner product

$$\langle f, g \rangle_{\mathcal{H}_{\rho,\alpha}} := \langle f, (\alpha \mathrm{Id} + K_\rho^{\otimes d})^{-1}[g] \rangle_{L_2(\rho\mathcal{X} \to \mathbb{R}^d)}. \tag{21}$$

It then determines a Riemannian metric on the function space. For $\tilde{p} \in \mathcal{H}_{\rho,\alpha}$ and $s = -\nabla \cdot (\rho \tilde{p})$, by (20) we have

$$\langle \nu_t, \tilde{p} \rangle_{\mathcal{H}_{\rho,\alpha}} = \mathbb{E}_{\rho_t(x)} \langle \nabla \log(\phi/\rho_t)(x), \tilde{p}(x) \rangle = -\int \log \frac{\phi}{\rho_t} (\nabla \cdot (\tilde{p}\rho)) dx = -(d\mathrm{KL}_\phi)(s),$$

i.e. with respect to the metric (21), SPOS is the gradient flow of the (negative) KL divergence functional.

Derivation of (19) let $(\lambda_i, \psi_i)_{i=1}^\infty$ be its eigendecomposition (i.e. the Mercer representation). For $j \in [d]$ let $\psi_{i,j} := \psi_i \mathbf{e}_j$ where $\{\mathbf{e}_j\}_{j=1}^d$ is the coordinate basis in $\mathbb{R}^d$, so $\{\lambda_i^{-1/2} \psi_{i,j}\}$ becomes an orthonormal basis in $\mathcal{H}^{\otimes d}$. Now we calculate the coordinate of $\phi^*_{\rho,\phi}$ in this basis.

$$\begin{aligned}
\langle \phi^*_{\rho,\phi}, \psi_{i,j} \rangle_{L_2(\rho)} &= \mathbb{E}_{\rho(x)} \mathbb{E}_{\rho(x')} \langle (\nabla_{x'} \log \phi(x')) k(x', x) + \nabla_{x'} k(x', x), \psi_{i,j}(x) \rangle \\
&= \mathbb{E}_{\rho(x')} \left[\langle \nabla_{x'} \log \phi(x'), (K_\rho[\psi_{i,j}])(x') \rangle + \nabla \cdot ((K_\rho[\psi_{i,j}])(x')) \right] \\
&=: \mathbb{E}_{\rho(x')}[S_\phi(K_\rho[\psi_{i,j}])(x')].
\end{aligned} \tag{22}$$

S_ϕ is known to satisfy the *Stein's identity*

$$\mathbb{E}_\rho S_\rho(g) = 0$$

for all $g \in \mathcal{H}$. Thus, we can subtract $\mathbb{E}_\rho S_\rho(K_\rho[\psi_{i,j}])$ from the right hand side of (22) without changing its value, and it becomes

$$\begin{aligned}
&\mathbb{E}_{\rho(x')}[S_\phi(K_\rho[\psi_{i,j}])(x')] - \mathbb{E}_{\rho(x')}[S_\rho(K_\rho[\psi_{i,j}])(x')] \\
&= \mathbb{E}_{\rho(x')} \left[\left\langle \nabla_{x'} \log \frac{\phi(x')}{\rho(x')}, (K_\rho[\psi_{i,j}])(x') \right\rangle \right] \\
&= \lambda_k \mathbb{E}_{\rho(x')} \left[\left\langle \nabla_{x'} \log \frac{\phi(x')}{\rho(x')}, \psi_{i,j}(x') \right\rangle \right].
\end{aligned}$$

As the equality holds for all i, k, we completed the derivation of (19).

E.3. MVL Objective Derived from SPOS

By (20) and (21), the MVL objective derived from SPOS is

$$\|\mathrm{grad}_\rho \mathrm{KL}_\phi\|^2_{\mathcal{H}_{\rho,\alpha}} = \langle \nabla \log(\phi/\rho_t), (\alpha \mathrm{Id} + K^{\otimes d}) \nabla \log(\phi/\rho_t) \rangle_{L_2(\rho\mathcal{X} \to \mathbb{R}^d)}.$$

In the right hand side above, the first term in the summation is the Fisher divergence, and the second is the kernelized Stein discrepancy (Liu et al., 2016b, Definition 3.2).

We note that a similar result for SVGD has been derived in (Liu and Wang, 2017), and our derivations connect to the observation that Langevin dynamics can be viewed as SVGD with a Dirac function kernel (thus SPOS also corresponds to SVGD with generalized-function-valued kernels).

E.4. Derivation of (8) in the Manifold Case

In this section we derive (8), when the latent-space distribution $q_\phi(z)$ is defined on a p-dimensional manifold embedded in some Euclidean space, and $H[q_\phi(z)]$ is the relative entropy w.r.t. the Hausdorff measure. The derivation is largely similar to the Euclidean case, and we only include it here for completeness.

(8) holds because

$$\begin{aligned}
\nabla_\phi \mathbb{H}\left[q_\phi(z)\right] &\overset{(i)}{=} -\nabla_\phi \mathbb{E}_{p(\epsilon)}\left[\log q_\phi\left(f(\epsilon,\phi)\right)\right] \\
&= -\mathbb{E}_{p(\epsilon)}\left[\nabla_\phi \log q_\phi\left(f(\epsilon,\phi)\right)\right] \\
&= -\mathbb{E}_{p(\epsilon)}\left[\nabla_\phi \log q_\phi(z)\big|_{z=f(\epsilon,\phi)} + \nabla_f \log q\left(f(\epsilon,\phi)\right)\nabla_\phi f(\epsilon,\phi)\right] \\
&\overset{(ii)}{=} -\mathbb{E}_{p(\epsilon)}\left[\nabla_z \log q_\phi(z) \nabla_\phi f(\epsilon,\phi)\right],
\end{aligned}$$

where (i) follows from Theorem 2.10.10 in Federer (2014), and (ii) follows from the same theorem as well as the fact that $\mathbb{E}_{q_\phi(z)}[\nabla_\phi \log q_\phi(z)] = \nabla_\phi \int q_\phi(z) dz = 0$.