Supplementary Material

A Derivation of (13)-(15)

Denote $\tilde{x} := x + \sqrt{\epsilon} z$.

$$\begin{aligned}
\hat{L}_{\rm dsm} &= \|x + \sqrt{\epsilon}z - \epsilon\nabla\mathcal{E}(x+\sqrt{\epsilon}z) - x\|^2 && (20)\\
&= \epsilon\|z\|^2 + \epsilon^2\|\nabla\mathcal{E}(\tilde{x})\|^2 - 2\epsilon^{3/2}\langle z, \nabla\mathcal{E}(\tilde{x})\rangle && (21)\\
&= \epsilon\|z\|^2 + \epsilon^2\|\nabla\mathcal{E}(\tilde{x})\|^2 - 2\epsilon^{3/2}\langle z, \nabla\mathcal{E}(x) + (\nabla^2\mathcal{E}(x))(\sqrt{\epsilon}z) + O(\epsilon)\rangle && (22)\\
&= \underbrace{\epsilon^2\left(\|\nabla\mathcal{E}(\tilde{x})\|^2 - 2z^\top(\nabla^2\mathcal{E}(x))z\right)}_{A} + \underbrace{\epsilon\|z\|^2 - 2\epsilon^{3/2}z^\top\nabla\mathcal{E}(x)}_{B} + o(\epsilon^2), && (23)
\end{aligned}$$

Notice

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

which is known as the Hutchinson's trick (Hutchinson, 1990), so $\lim_{\epsilon\to 0}\epsilon^{-2}\mathbb{E}(A)$ is two times the Fisher divergence $D_{\rm F}(p|q)$. But $Var(B) = O(\epsilon^2)$, so as $\epsilon \to 0$, the rescaled estimator $\epsilon^{-2}\hat{L}_{\rm dsm}$ becomes unbiased with *infinite variance*; and subtracting (B) from (A) results in a finite-variance estimator.

B 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 $(d\mathrm{KL}_\phi)_\rho(s) := \int s(x)\log\frac{\rho(x)}{\phi(x)}dx$.

B.1 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

$$\phi^*_{\rho,\phi} = K_\rho^{\otimes d}[\nabla\log(\phi/\rho)], \tag{24}$$

which we will derive at the end of this subsection for completeness. Following (24) we have

$$\nu_t(x) = (\alpha\mathrm{Id} + K_\rho^{\otimes d})[\nabla\log(\phi/\rho)]. \tag{25}$$

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

$$\langle f, g\rangle_{\mathcal{H}_{\rho,\alpha}} := \langle f, (\alpha\mathrm{Id} + K_\rho^{\otimes d})^{-1}[g]\rangle_{L_2(\rho\mathcal{X}\to\mathbb{R}^d)}. \tag{26}$$

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 (25) 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 (26), SPOS is the gradient flow minimizing the KL divergence functional.

Derivation of (24) 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.

$$\begin{aligned}\langle \phi^*_{\rho,\phi}, \psi_{i,j}\rangle_{L_2(\rho)} &= \mathbb{E}_{\rho(x)}\mathbb{E}_{\rho(x')}\langle(\nabla_{x'}\log\phi(x'))k(x',x) + \nabla_{x'}k(x',x), \psi_{i,j}(x)\rangle \\ &= \mathbb{E}_{\rho(x')}\left[\langle\nabla_{x'}\log\phi(x'), (K_\rho[\psi_{i,j}])(x')\rangle + \nabla\cdot((K_\rho[\psi_{i,j}])(x'))\right] \\ &=: \mathbb{E}_{\rho(x')}[S_\phi(K_\rho[\psi_{i,j}])(x')].\end{aligned} \tag{27}$$

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 (27) 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 (24).

B.2 MVL Objective Derived from SPOS

By (25) and (26), 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).

C Justification of the Use of Local Coordinates in (17)

In this section, we prove in Proposition C.1 that the local coordinate representation lead to valid approximation to the MVL objective in the compact case. We also argue in Remark C.2 that the use of local coordinate does not lead to numerical instability.

Remark C.1. *While a result more general than Proposition C.1 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 C.1. ***(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*

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

for any $L \ge 1$.

Proposition C.1. *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* (17)*, X_t following the true Riemannian Langevin dynamics targeting $q^{1/2}$. Then*

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

i.e. (17) *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 C.1. 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.[7] By Hsu (2008), until τ, $\bar{X}_t$ follows the local coordinate representation of Riemannian LD (3), thus on the event $\{\epsilon \le \tau\}$, $\bar{X}_\epsilon$ would correspond to y^- in (18). As $\mathcal{X}$ is compact, the continuous energy function $\mathcal{E}$ is bounded by $|\mathcal{E}(\cdot)| \le A$ for some finite A. Then for sufficiently small ϵ,

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

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 $\left|\frac{\mathbb{E}(-\mathcal{E}(X_\epsilon)\mathbf{1}_{\{\tau\le\epsilon\}})}{\epsilon}\right| \le \frac{A\mathbb{P}(\tau\le\epsilon)}{\epsilon} = \frac{A\mathbb{P}(\bar{\tau}\le\epsilon)}{\epsilon} \le \frac{Ae^{-C/2\epsilon}}{\epsilon} \to 0$ when $\epsilon \to 0$. Hence the proof is complete. □

Remark C.2. *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.*

D Derivation of (19) in the Manifold Case

In this section we derive (19), 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.

(19) holds because

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

[7] 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 $\tau = \infty$.

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

E Experiment Details and Additional Results

Code will be available at `https://github.com/thu-ml/wmvl`.

E.1 Synthetic Experiments

E.1.1 Experiment Details

Experiment Details in Section 6.1.1 The (squared) bias is estimated as follows: denote the SSM estimator and ours as $\mathbb{E}_{p(x)\mathcal{N}(\epsilon|0,1)}[L_F^{\mathrm{ssm}}(x;\epsilon)]$ and $\mathbb{E}_{p(x)\mathcal{N}(\epsilon|0,1)}[L_F^{\mathrm{mvl}}(x;\epsilon)]$, respectively. One could verify that both methods estimate (7). Our estimate for the squared bias is now $\frac{1}{K}\sum_{k=1}^{K}\left(\frac{1}{M}\sum_{j=1}^{M}(L_F^{\mathrm{ssm}}(x^{(k)};\epsilon^{(j)}) - L_F^{\mathrm{mvl}}(x^{(k)};\epsilon^{(j)}))\right)^2$. where $x(k) \sim p(x), \epsilon^{(j)} \sim \mathcal{N}(0,1)$ are i.i.d. draws. The expectation of this estimate upper bounds the true squared bias by Cauchy's inequality, and the bias → 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.

The same procedure is applied to the denoising score matching estimator.

Experiment Details in Section 6.1.2 For this experiment, the data distribution is chosen as

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

where p_{vM} is the von Mises density

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

For the model distribution, the energy function 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}.

E.1.2 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 6.1.1. The results are shown in Figure 3. We can see that with the introduction of the control variate, CD-1 performs as well as other score matching methods.

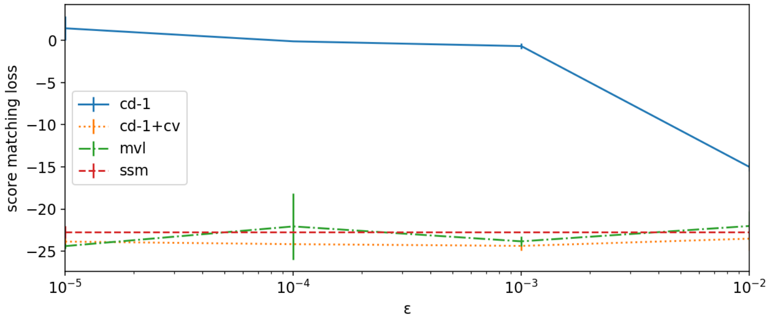


Figure 3: Score matching loss for different methods, with varying step-size. Lower is better.

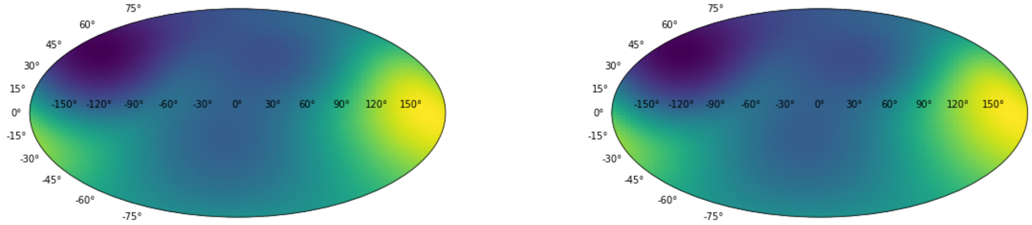


Figure 4: Mollweide projections of the ground truth (left) and learnt (right) energy functions on S^2.

E.1.3 Learning EBMs on S^2

As a slightly more involved test case for our Riemannian score matching approximation, we consider learning EBMs on S^2. The target distribution is a mixture of 4 von-Mises-Fisher distributions. The ground truth and learnt energy functions are plotted in Figure 4; we can see that our method leads to a good fit.

E.2 Auto-Encoder Experiments

In all auto-encoder experiments, setup follows from (Song et al., 2019) whenever possible. The only difference is that for score estimation, we parameterize the energy function, and use 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 also noted that in 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 E.1.1, 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^{-3}, and apply L2 regularization on the energy scale with coefficient 10^{-5}. For the WAE-GAN baseline, we parameterize the GAN discriminator as a feed-forward network with 2 hidden layers, each with 256 units. We use tanh activation, and apply L2 regularization with coefficient 10^{-5}. All models are trained for 200,000 iterations using RMSProp, using 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. FID scores are calculated using the implementation in (Heusel et al., 2017).

Sampled Generations in the Auto-encoder Experiments See Figure 7 - 9.

E.2.1 WAE Experiments in Higher Dimensions

In this section, we present results of hyperspherical WAEs on CIFAR-10 and CelebA, with larger n_z.

For CelebA we follow the setup in Song et al. (2019): $n_z = 32$, RMSProp, learning rate 10^{-4}, train for 100,000 iterations. In addition, we apply spectral normalization and L2 regularization with coefficient 10^{-4}. The step-size in the MVL approximation is set to 10^{-4}. The FID scores, averaged over 5 runs, are 50.82 ± 0.50 for our method and 51.20 ± 0.59 for WAE-GAN.

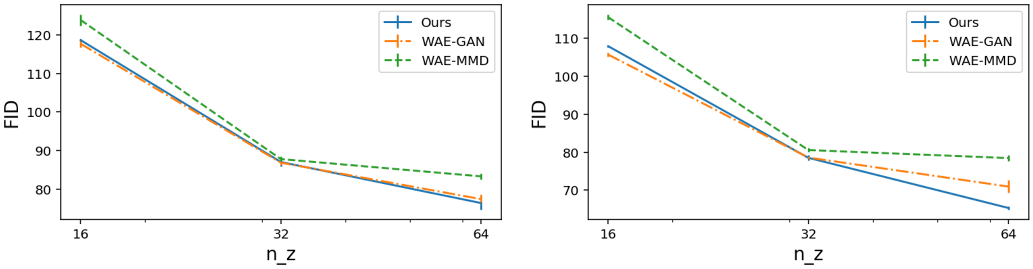
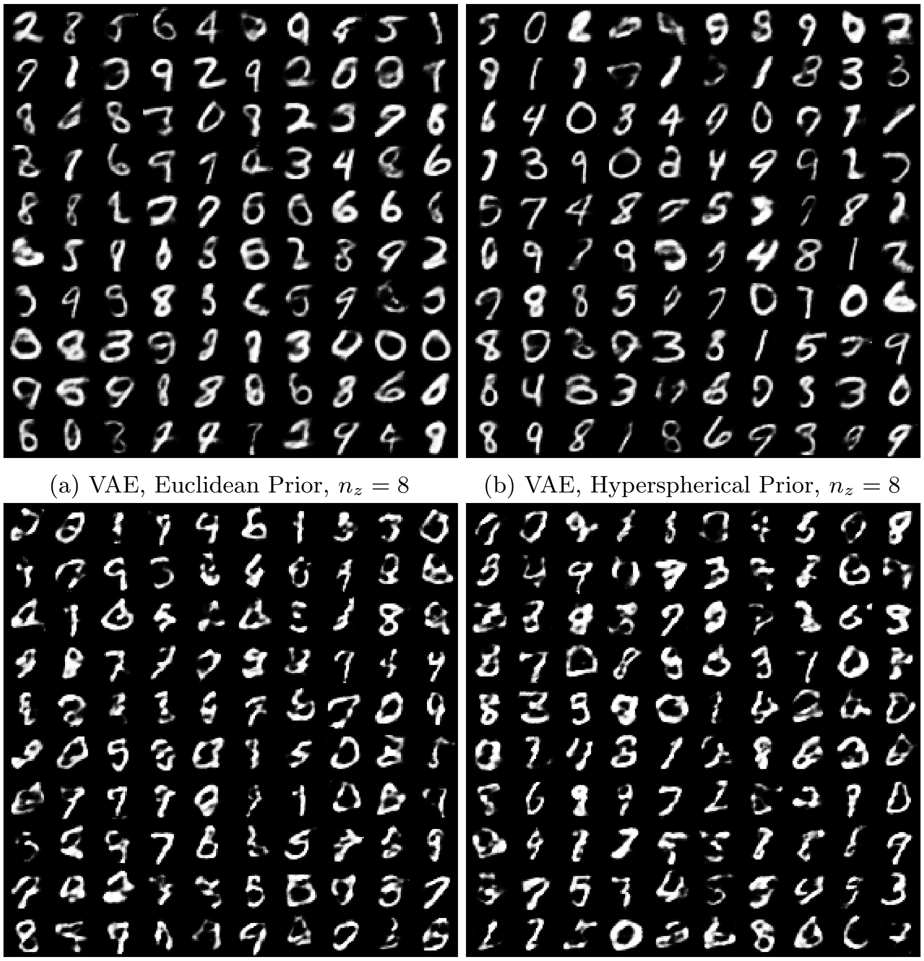


Figure 5: FID on CIFAR-10, with varying n_z. Left: after 10^5 iterations; right: after 2×10^5 iterations.

For CIFAR-10, we modify the auto-encoder architecture and remove one scaling block to account for its lower resolution. We do not use spectral normalization which leads to slightly worse results. The FID scores for varying n_z are presented in Figure 5, where we can see our method compares favorably to all baselines.



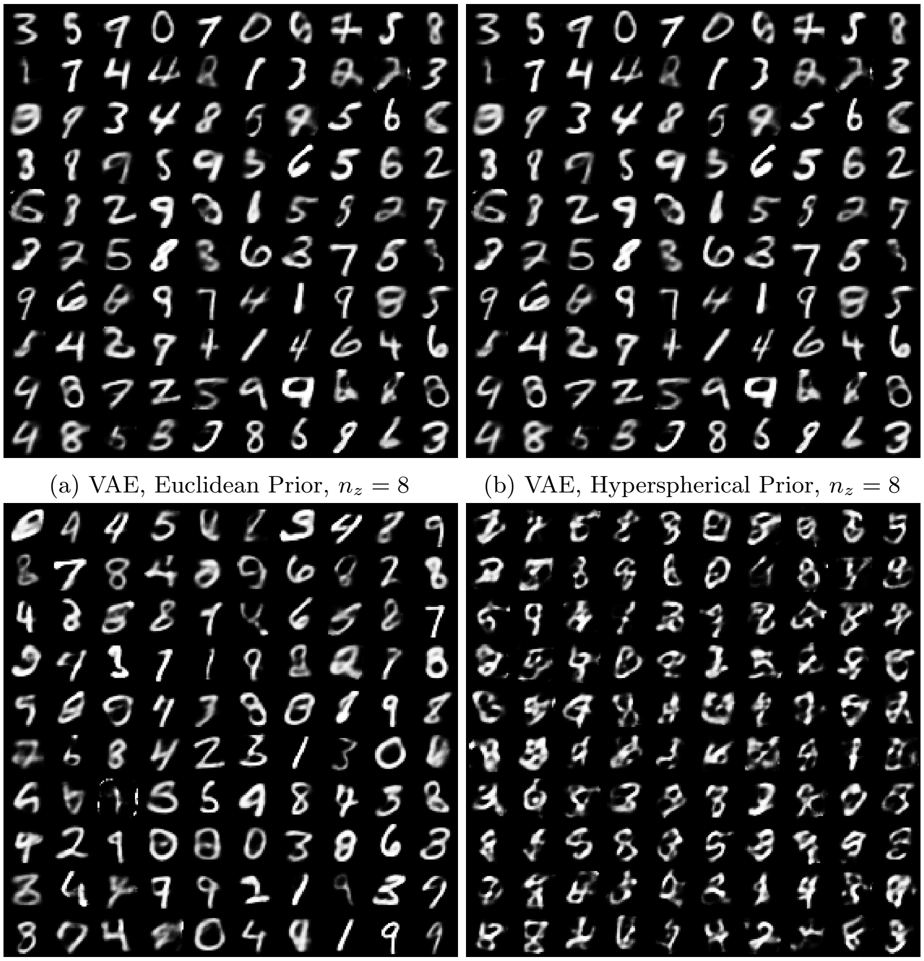
(a) VAE, Euclidean Prior, $n_z = 8$

(b) VAE, Hyperspherical Prior, $n_z = 8$

(c) VAE, Euclidean Prior, $n_z = 32$

(d) VAE, Hyperspherical Prior, $n_z = 32$

Figure 6: Sampled generations of **implicit** VAEs.



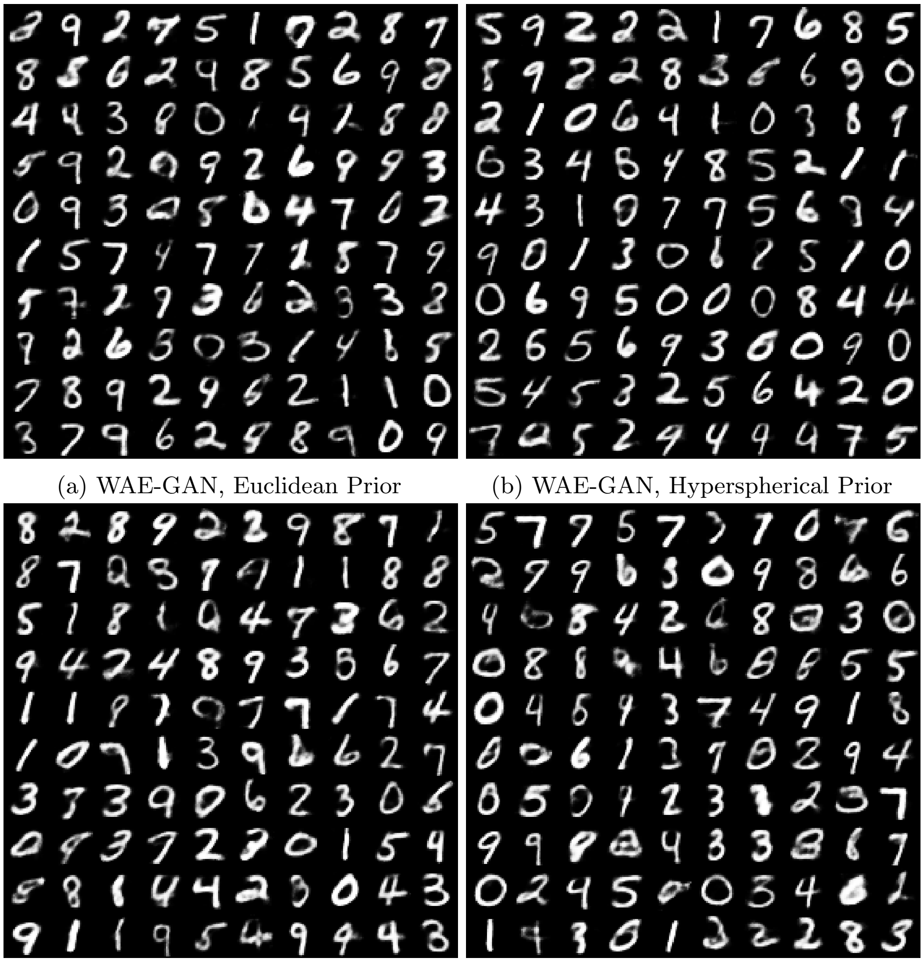
(a) VAE, Euclidean Prior, $n_z = 8$

(b) VAE, Hyperspherical Prior, $n_z = 8$

(c) VAE, Euclidean Prior, $n_z = 32$

(d) VAE, Hyperspherical Prior, $n_z = 32$

Figure 7: Sampled generations of **explicit** VAEs.



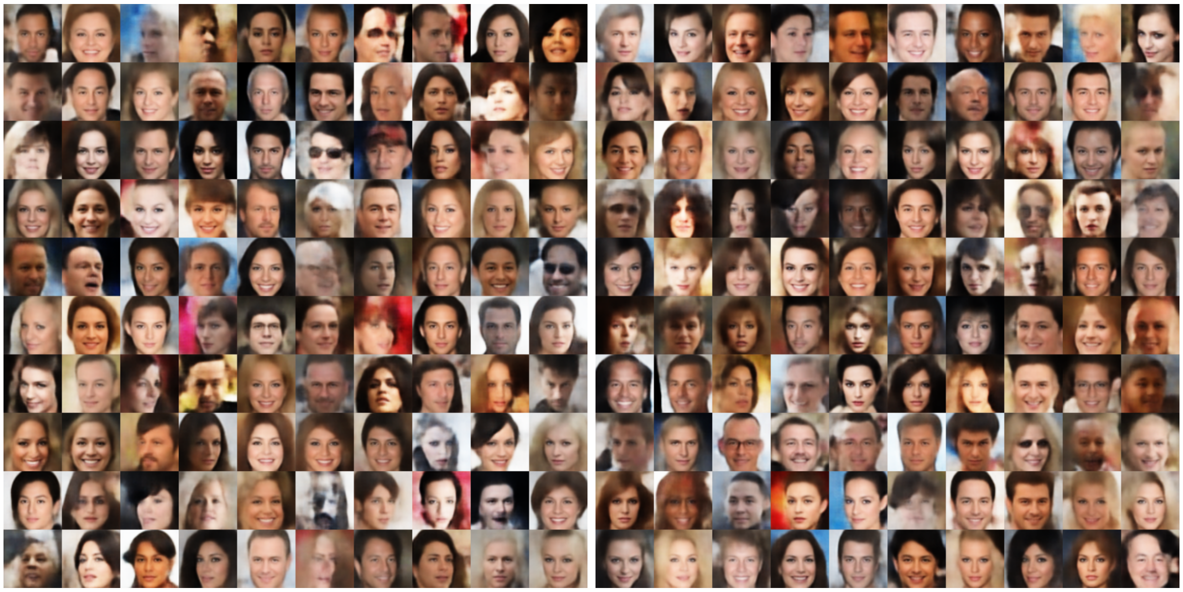
(a) WAE-GAN, Euclidean Prior

(b) WAE-GAN, Hyperspherical Prior

(c) WAE-MVL, Euclidean Prior

(d) WAE-MVL, Hyperspherical Prior

Figure 8: Sampled generations in the WAE experiment on MNIST.



(a) WAE-GAN, Hyperspherical Prior

(b) WAE-MVL, Hyperspherical Prior

Figure 9: Sampled generations in the WAE experiment on CelebA.