2nd Symposium on Advances in Approximate Bayesian Inference, 2019 1–17

Towards Characterizing the High-dimensional Bias of Kernel-based Particle Inference Algorithms

Jimmy Ba JBA@CS.TORONTO.EDU
University of Toronto & Vector Institute for Artificial Intelligence

Murat A. Erdogdu ERDOGDU@CS.TORONTO.EDU
University of Toronto & Vector Institute for Artificial Intelligence

Marzyeh Ghassemi MARZYEH@CS.TORONTO.EDU
University of Toronto & Vector Institute for Artificial Intelligence

Taiji Suzuki TAIJI@MIST.I.U-TOKYO.AC.JP
University of Tokyo & RIKEN AIP

Denny Wu DENNYWU@CS.TORONTO.EDU
University of Toronto & Vector Institute for Artificial Intelligence

Shengyang Sun SSY@CS.TORONTO.EDU
University of Toronto & Vector Institute for Artificial Intelligence

Tianzong Zhang ZTZ16@MAILS.TSINGHUA.EDU.CN
Tsinghua University & Vector Institute for Artificial Intelligence

Abstract

Particle-based inference algorithm is a promising method to efficiently generate samples for an intractable target distribution by iteratively updating a set of particles. As a noticeable example, Stein variational gradient descent (SVGD) provides a deterministic and computationally efficient update, but it is known to underestimate the variance in high dimensions, the mechanism of which is poorly understood. In this work we explore a connection between SVGD and MMD-based inference algorithm via Stein's lemma. By comparing the two update rules, we identify the source of bias in SVGD as a combination of high variance and *deterministic bias*, and empirically demonstrate that the removal of either factors leads to accurate estimation. In addition, for learning high-dimensional Gaussian target, we analytically derive the converged variance for both algorithms, and confirm that only SVGD suffers from the curse of dimensionality.

1. Introduction

The Stein Variational Gradient Descent (SVGD) (Liu and Wang, 2016) is a deterministic particle-based inference algorithm that iteratively transports the particles by the functional gradient in the reproducing kernel Hilbert space (RKHS) of KL-divergence, which takes the form of a kernelized Stein's operator. In contrast to the empirical successes (Liu et al., 2017; Haarnoja et al., 2017; Kim et al., 2018), very few convergence guarantees have been established for SVGD except for the mean-field regime (Liu and Wang, 2018; Lu et al., 2019). Moreover, it has been observed that the variance estimated by SVGD scales inversely with the dimensionality of the problem. This is a highly undesirable property for two

© J. Ba, M.A. Erdogdu, M. Ghassemi, T. Suzuki, D. Wu, S. Sun & T. Zhang.

reasons: 1) underestimating the variance leads to failures of explaining the uncertainty of model predictions; 2) modern inference problems are usually high-dimensional. For example, Bayesian neural networks (MacKay, 1992) could be more than millions of dimensions.

We study the algorithmic bias of SVGD that leads to the variance underestimation in high dimensions. We construct another kernel-based inference algorithm termed *MMD-descent*, which closely resembles SVGD but estimate the variance accurately. By comparing their updates, we identify the cause of variance collapse in SVGD as a combination of high variance due to Stein's lemma, and *deterministic bias*, i.e. the inability to resample particles. We empirically verify that removing either of these two factors, while computationally expensive, leads to accurate variance estimation. Then, under mild assumptions, we derive the equilibrium variance of SVGD and MMD-descent in matching high-dimensional Gaussians, and confirm that variance estimated by SVGD scales inversely with the dimensionality.

2. Connecting SVGD and MMD-Descent

Particle variational inference approximates an intractable distribution $p(\boldsymbol{x})$ with a set of particles $X = \{\boldsymbol{x}_i\}_{i=1}^n$. Specifically, we iteratively optimize a set of particles $X = \{\boldsymbol{x}_i\}_{i=1}^n$ under the deterministic update: $\boldsymbol{x}_i = \boldsymbol{x}_i + \epsilon\Delta(\boldsymbol{x}_i)$, where ϵ is the stepsize, and $\Delta(\cdot) : \mathbb{R}^d \to \mathbb{R}^d$ represents the update direction. SVGD defines the update direction as

$$\Delta^{\text{SVGD}}(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^{n}[\underbrace{\nabla_{\boldsymbol{x}_i}\log p(\boldsymbol{x}_i)k(\boldsymbol{x}_i,\boldsymbol{x})}_{\text{driving force}} + \underbrace{\nabla_{\boldsymbol{x}_i}k(\boldsymbol{x}_i,\boldsymbol{x})}_{\text{repulsive force}}] := \frac{1}{n}\sum_{i=1}^{n}[S_1(\boldsymbol{x}_i,\boldsymbol{x}) + S_2(\boldsymbol{x}_i,\boldsymbol{x})], \quad (1)$$

where k is a positive definite kernel. Intuitively, the *log derivative* term $S_1(\boldsymbol{x}_i, \boldsymbol{x})$ corresponds to a *driving force* that guides particles towards high likelihood regions, whereas the *kernel derivative* term $S_2(\boldsymbol{x}_i, \boldsymbol{x})$ provides a *repulsive force* to prevent the particles from collapsing.

We now introduce another particle inference algorithm MMD-descent, motivated from kernel herding (Welling, 2009). Instead of selecting particles one by one to minimize the mean maximum discrepancy (MMD) (Gretton et al., 2012) with respect to the target, we jointly optimize all particles together to minimize MMD via gradient descent. For symmetric kernel, such as the Euclidean distance kernel (Definition 3), the update can be written as:

$$\Delta^{\text{MMD}}(\boldsymbol{x}) = \mathbb{E}_{\boldsymbol{y}\sim p}\underbrace{[-\nabla_{\boldsymbol{y}}k(\boldsymbol{x},\boldsymbol{y})]}_{\text{driving force}} + \frac{1}{n}\sum_{i=1}^{n}\underbrace{\nabla_{\boldsymbol{x}_i}k(\boldsymbol{x}_i,\boldsymbol{x})}_{\text{repulsive force}} = -\mathbb{E}_{\boldsymbol{y}\sim p}[S_2(\boldsymbol{y},\boldsymbol{x})] + \frac{1}{n}\sum_{i=1}^{n}S_2(\boldsymbol{x}_i,\boldsymbol{x}). \quad (2)$$

Note that MMD-descent is not practical since integration under the target distribution is usually infeasible. When the kernel k is in the Stein class of p, we have the equivalence:

$$-\mathbb{E}_{\boldsymbol{y}\sim p}[S_2(\boldsymbol{y},\boldsymbol{x})] = -\mathbb{E}_{\boldsymbol{y}\sim p}[\nabla_{\boldsymbol{y}}k(\boldsymbol{x},\boldsymbol{y})] = \mathbb{E}_{\boldsymbol{y}\sim p}[\nabla_{\boldsymbol{y}}\log p(\boldsymbol{y})k(\boldsymbol{x},\boldsymbol{y})] = \mathbb{E}_{\boldsymbol{y}\sim p}[S_1(\boldsymbol{y},\boldsymbol{x})], \quad (3)$$

SVGD vs. MMD-descent. Observe that 1) the repulsive force term in SVGD and MMD-descent is identical; 2) in MMD-descent, the driving force is integrated under the target distribution p, whereas in SVGD under the current particle distribution q.

It is clear that at the infinite particle limit, $q = p$ is the fixed point for both updates. However, such asymptotic property does not entail 1) the two algorithms reliably approximate the target distribution in high dimensions, i.e. when n, d are both large ; 2) the two

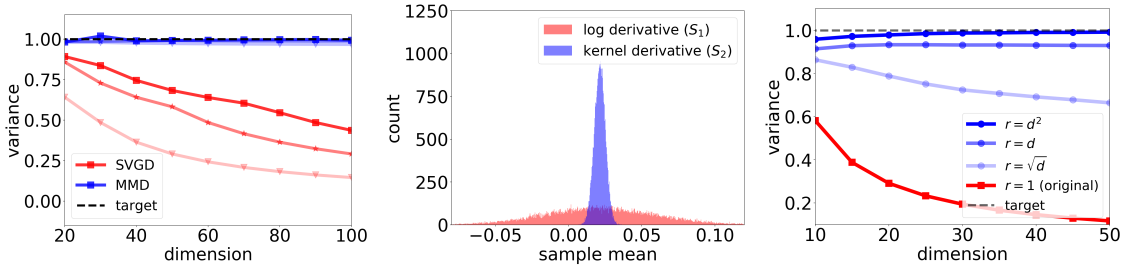
algorithms converges to similar stationary points under finite samples. Given the similar updates, it is natural to ask: do SVGD and MMD-descent approximate the target distribution reliably in high dimensions and are their approximations similar?

The answer is in the negative: SVGD and MMD-descent converge to different stationary points. Specifically, SVGD underestimates the marginal variance in high dimensions (Zhuo et al., 2017). For unit Gaussian targets, although both algorithms correctly estimates the mean, Figure 1(a) illustrates that SVGD particles have decreasing marginal variance as dimensionality increases whereas MMD-descent particles approximate the marginal variance accurately. In the following we characterize the sources of this pitfall of SVGD.

3. Understanding the Pitfall of SVGD

Variance from Integration by Parts The convergence of SVGD crucially depends on the equality $\mathbb{E}_{\boldsymbol{y}\sim p}[S_1(\boldsymbol{y},\boldsymbol{x})] = -\mathbb{E}_{\boldsymbol{y}\sim p}[S_2(\boldsymbol{y},\boldsymbol{x})]$ obtained via integration by parts. Nevertheless, the variance of S_1 and S_2 may differ drastically, hence invoking convergence issues under finite particles. In general, S_1 can have much larger magnitude, resulting in large variance of the driving force term in SVGD. In contrast, in MMD-descent the driving force can be computed via integration S_2, and thus the high variance term is not involved. We visualize the difference in the variance of estimating S_1 and S_2 in Figure 1(b), and provide the following characterization for the Gaussian RBF kernel and Gaussian target.

Proposition 1 *Define the mean squared error as:* $\mathrm{MSE}_p[f(\boldsymbol{y})] = \mathbb{E}_{p(\boldsymbol{y})} \|f(\boldsymbol{y}) - \mathbb{E}_p[f(\boldsymbol{y})]\|_2^2$. *Then for* $p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}\,|\,\mu, I_d)$, *Gaussian kernel* $k(\boldsymbol{x},\boldsymbol{y}) = \exp\left(\sigma^{-2}\|\boldsymbol{x}-\boldsymbol{y}\|_2^2/2\right)$ *with* $\sigma \in \Theta(\sqrt{d})$, *and* $\boldsymbol{x} \in \mathbb{S}^{d-1}(\sqrt{d})$, *we have* $\mathrm{MSE}_p[S_2(\boldsymbol{y},\boldsymbol{x})] \in \Theta(d^{-1})$; $\mathrm{MSE}_p[S_1(\boldsymbol{y},\boldsymbol{x})] \in \Theta(d)$.



(a) marginal variance. (b) distribution of S_1 and S_2. (c) variance of SVGD-VR.

Figure 1: (a) dimension-averaged marginal variance of converged particles under SVGD and MMD-descent. Darker color represents larger number of particles. (b) 1D visualization of the distribution of sample means of the two estimators S_1 and S_2 under 10-dimensional Gaussian target. (c) marginal variance of particles obtained from SVGD using r-times more samples to estimate S_1 and thus reduce variance. We set $n = 10$ and vary d.

High variance of S_1 entails that larger number of samples is required to accurately estimate S_1. Due to this discrepancy, we expect SVGD to better approximate the target if more samples are used to estimate the driving force. Assume we keep rn $(r > 1)$ particles,

use all of them to estimate S_1 and n particles for S_2. The modified update is given as

$$\Delta^{\text{SVGD-VR}}(\boldsymbol{x}) = \frac{1}{rn}\sum_{i=1}^{rn} S_1(\boldsymbol{x}_i, \boldsymbol{x}) + \frac{1}{n}\sum_{i=1}^{n} S_2(\boldsymbol{x}'_i, \boldsymbol{x}).$$

Where $\{\boldsymbol{x}'_i\}_{i=1}^n$ are drawn randomly from $\{\boldsymbol{x}_i\}_{i=1}^{rn}$. This is to say, at each step the transport map is constructed via estimating the repulsive force with n particles and the driving force with rn particles. As shown in Figure 1(b), even though the repulsive force S_2 is calculated with few samples, when $r \in \Omega(d)$ the algorithm accurately estimates the variance (independent of d). We comment that although the modification corrects the variance collapse, it is not practical since the required number of particles scales with the dimensionality.

Bias from Deterministic Update. The analysis above suggests that the pitfall of SVGD relates to the high variance of the driving force S_1, which is not present in MMD-descent. However in scenarios like gradient estimation of variational inference, high variance usually results in slower convergence, but not necessarily variance collapse. In SVGD, the particles $\{\boldsymbol{x}_i\}_{i=1}^n$ used to compute the update are assumed as random samples from an underlying continuous distribution. However, due to the deterministic update, the distribution q is entirely represented by the same set of particles and drawing random samples is not possible. We now demonstrate that this *deterministic bias*, when combined with high variance estimators, may cause the algorithm to converge to biased target or even diverge.

We start from an illustrative experiment of deterministic bias with MMD-descent. Given target samples $\{\boldsymbol{y}_i\}_{i=1}^m \sim p(y)$, we have two forms of update that differs in the driving force.

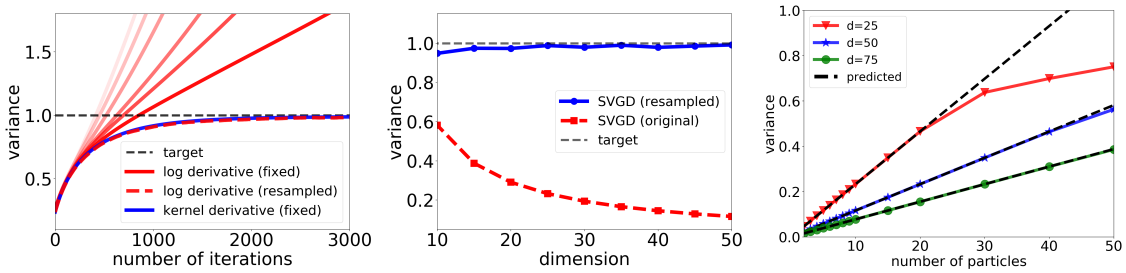
$$\Delta_1^{\text{MMD}}(\boldsymbol{x}) = \frac{1}{m}\sum_{i=1}^{m} S_1(\boldsymbol{y}_i, \boldsymbol{x}) + \frac{1}{n}\sum_{i=1}^{n} S_2(\boldsymbol{x}_i, \boldsymbol{x});\ \ \Delta_2^{\text{MMD}}(\boldsymbol{x}) = -\frac{1}{m}\sum_{i=1}^{m} S_2(\boldsymbol{y}_i, \boldsymbol{x}) + \frac{1}{n}\sum_{i=1}^{n} S_2(\boldsymbol{x}_i, \boldsymbol{x}).$$

As argued above Δ_1^{MMD} tends to have higher variance due to S_1. Note that the estimation of $\mathbb{E}_{p(\boldsymbol{y})}[S_2(\boldsymbol{y}, \boldsymbol{x})]$ is unbiased when $\boldsymbol{y}_i$ is resampled at each iteration, and empirically the converged variance is unbiased indeed (*log derivative (resampled)* in Figure 2(a)).

To simulate the deterministic bias, we sample $\{\boldsymbol{y}_i\}_{i=1}^m \sim p(\boldsymbol{y})$ and keep them $fixed$ throughout optimization. As shown in Figure 2(a), Δ_2^{MMD} can estimate the variance accurately (*kernel derivative (fixed)*), but Δ_1^{MMD} diverges (*log derivative (fixed)*). As expected, the deterministic bias of estimating S_1 is more significant in Δ_1^{MMD} due to high variance.

This experiment shows that the deterministic bias in SVGD arises from the algorithm not being able to resample from q. We now design an algorithm to achieve random "resampling". Let q_0 be the continuous distribution where initial samples are drawn from. Let the particles at t-th iteration be $\hat{q}_t$. We randomly draw new samples $\hat{q}'_0$ from q_0. At the i-th iteration, we update $\hat{q}'_i$ with Δ^{SVGD} using the map defined by particles $\hat{q}_i$. Because both $\hat{q}_T$ and $\hat{q}'_T$ are initially sampled from q_0 and transported using the same map defined by $\{\hat{q}_i\}_{i=0}^{T-1}$, $\hat{q}_T$ and $\hat{q}'_T$ has the same distribution. So $\hat{q}'_T$ can be seen as "resampled" from the same distribution as $\hat{q}_T$, and we can use $\hat{q}'_T$ for updating $\hat{q}_T$ in SVGD without the deterministic bias. The algorithm is reminiscent of flow-based variational inference (Rezende and Mohamed, 2015) and transport-based particle gradient descent (Nitanda and Suzuki, 2017) algorithms.

As shown in Figure 2(b), SVGD with this resampling scheme accurately estimates the target variance with a small number of particles being updated at each iteration ($n = 10$).



(a) variations of MMD-descent. (b) SVGD-resampled. (c) SVGD.

Figure 2: (a) marginal variance of converged particles obtained via two forms of MMD-descent updates. Darker color indicates a larger number of particles (from 50 to 3000). Observe that resampling or Δ_2^{MMD} with fixed samples converges, whereas Δ_1^{MMD} with fixed samples diverges. (b) SVGD with resampling scheme correctly estimates the variance. We set $n = 10$. (c) marginal variance of particles converged under SVGD in learning unit Gaussian distribution predicted by Proposition 2.

We expect similar outcomes in estimating higher order moments since the algorithm is completely *unbiased*. But the computational cost of such resampled updates scales quadratically with the number of iterations, thus rendering the method impractical in real applications.

Analytically Deriving the Variance. We now quantitatively characterize the variance collapse in SVGD by deriving the variance of the converged particles in learning a unit Gaussian in high dimensions. Specifically, we consider the setup where n, d tend to infinity at the same rate. Various works have shown that in this regime the kernel matrix can be asymptotically decomposed into a weighted sum of the data covariance matrix and a scaled identity (El Karoui et al., 2010; Cheng and Singer, 2013; Bordenave et al., 2013). We perform a similar decomposition via Taylor expansion and obtain the following characterization:

Proposition 2 (Informal) *For unit Gaussian target and Gaussian RBF kernel, assume that particles at the fixed point of both algorithms correlate weakly and have concentrated norm, then as* $d,n \to \infty$ *and* $\lim_{n\to\infty} n/d = \gamma \in (0,1)$*, particles driven by SVGD* (7) *equilibrates at the marginal variance* $v^{SVGD} \to \frac{1}{e-1}\gamma$*, whereas MMD-descent* (2) *leads to* $v^{MMD} \to 1$*.*

Empirical results in Figure 2(c) align with the prediction: when $d > n$, the equilibrium variance of SVGD scales linearly with n but is also inverse to d. This indicates that as the dimensionality increases, more particles is required to reliably estimate the true variance. When $\gamma > 1$, the variance empirically approaches the target variance from below as γ increases. On the other hand, in this regime MMD-descent does not underestimate the variance for all γ.

References

Francis Bach, Simon Lacoste-Julien, and Guillaume Obozinski. On the equivalence between herding and conditional gradient algorithms. *arXiv preprint arXiv:1203.4523*, 2012.

Andrew D Barbour. Stein's method and poisson process convergence. *Journal of Applied Probability*, 25(A):175–184, 1988.

Charles Bordenave et al. On euclidean random matrices in high dimension. *Electronic Communications in Probability*, 18, 2013.

Wilson Ye Chen, Lester Mackey, Jackson Gorham, François-Xavier Briol, and Chris J Oates. Stein points. *arXiv preprint arXiv:1803.10161*, 2018.

Xiuyuan Cheng and Amit Singer. The spectrum of random inner-product kernel matrices. *Random Matrices: Theory and Applications*, 2(04):1350010, 2013.

Kacper Chwialkowski, Heiko Strathmann, and Arthur Gretton. A kernel test of goodness of fit. JMLR: Workshop and Conference Proceedings, 2016.

Noureddine El Karoui et al. The spectrum of kernel random matrices. *The Annals of Statistics*, 38(1):1–50, 2010.

Murat A Erdogdu. Newton-stein method: an optimization method for glms via stein's lemma. *The Journal of Machine Learning Research*, 17(1):7565–7616, 2016.

Murat A Erdogdu, Mohsen Bayati, and Lee H Dicker. Scalable approximations for generalized linear problems. *arXiv preprint arXiv:1611.06686*, 2016.

Chengyue Gong, Jian Peng, and Qiang Liu. Quantile stein variational gradient descent for batch bayesian optimization. In *International Conference on Machine Learning*, pages 2347–2356, 2019.

Jackson Gorham and Lester Mackey. Measuring sample quality with stein's method. In *Advances in Neural Information Processing Systems*, pages 226–234, 2015.

Jackson Gorham and Lester Mackey. Measuring sample quality with kernels. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1292–1301. JMLR. org, 2017.

Jackson Gorham, Andrew B Duncan, Sebastian J Vollmer, and Lester Mackey. Measuring sample quality with diffusions. *arXiv preprint arXiv:1611.06972*, 2016.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar):723–773, 2012.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1352–1361. JMLR. org, 2017.

Ferenc Huszár. Variational inference using implicit distributions. *arXiv preprint arXiv:1702.08235*, 2017.

Ferenc Huszár and David Duvenaud. Optimally-weighted herding is bayesian quadrature. *arXiv preprint arXiv:1204.1664*, 2012.

Taesup Kim, Jaesik Yoon, Ousmane Dia, Sungwoong Kim, Yoshua Bengio, and Sungjin Ahn. Bayesian model-agnostic meta-learning. *arXiv preprint arXiv:1806.03836*, 2018.

Yingzhen Li and Richard E Turner. Gradient estimators for implicit models. *arXiv preprint arXiv:1705.07107*, 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. Stein variational gradient descent as moment matching. In *Advances in Neural Information Processing Systems*, pages 8868–8877, 2018.

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, 2016.

Yang Liu, Prajit Ramachandran, Qiang Liu, and Jian Peng. Stein variational policy gradient. *arXiv preprint arXiv:1704.02399*, 2017.

Jianfeng Lu, Yulong Lu, and James Nolen. Scaling limit of the stein variational gradient descent: The mean field regime. *SIAM Journal on Mathematical Analysis*, 51(2):648–671, 2019.

David JC MacKay. A practical Bayesian framework for backpropagation networks. *Neural Computation*, 4:448–472, 1992.

Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997.

Atsushi Nitanda and Taiji Suzuki. Stochastic particle gradient descent for infinite ensembles. *arXiv preprint arXiv:1712.05438*, 2017.

Chris J Oates, Jon Cockayne, François-Xavier Briol, and Mark Girolami. Convergence rates for a class of estimators based on stein's method. *arXiv preprint arXiv:1603.03220*, 2016.

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

Jiaxin Shi, Shengyang Sun, and Jun Zhu. A spectral approach to gradient estimation for implicit distributions. *arXiv preprint arXiv:1806.02925*, 2018.

Bharath K Sriperumbudur, Kenji Fukumizu, and Gert RG Lanckriet. Universality, characteristic kernels and rkhs embedding of measures. *Journal of Machine Learning Research*, 12(Jul):2389–2410, 2011.

Charles Stein et al. A bound for the error in the normal approximation to the distribution of a sum of dependent random variables. In *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Volume 2: Probability Theory*. The Regents of the University of California, 1972.

Richard S Sutton, Andrew G Barto, et al. *Introduction to reinforcement learning*, volume 2. MIT press Cambridge, 1998.

Dilin Wang and Qiang Liu. Nonlinear stein variational gradient descent for learning diversified mixture models. In *International Conference on Machine Learning*, pages 6576–6585, 2019.

Dilin Wang, Zhe Zeng, and Qiang Liu. Stein variational message passing for continuous graphical models. In *International Conference on Machine Learning*, pages 5206–5214, 2018.

Max Welling. Herding dynamic weights for partially observed random field models. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 599–606. AUAI Press, 2009.

Jaesik Yoon, Taesup Kim, Ousmane Dia, Sungwoong Kim, Yoshua Bengio, and Sungjin Ahn. Bayesian model-agnostic meta-learning. In *Advances in Neural Information Processing Systems*, pages 7332–7342, 2018.

Jingwei Zhuo, Chang Liu, Jiaxin Shi, Jun Zhu, Ning Chen, and Bo Zhang. Message passing stein variational gradient descent. *arXiv preprint arXiv:1711.04425*, 2017.

Appendix A. Additional Information

A.1. Background on SVGD and MMD-descent

Integral Probability Metric

To measuring how well a set of samples approximates a target distribution p, one may consider the maximum discrepancy between the target p and sample distribution q over some function class $\mathcal{F}$:

$$D_{\mathcal{F}}(p, q) = \sup_{f \in \mathcal{F}} \mathbb{E}_q[f(\boldsymbol{x})] - \mathbb{E}_p[f(\boldsymbol{y})],$$

which is known as the *integral probability metric* (IPM) (Müller, 1997). In particular, if $\mathcal{F}$ is a unit ball in the reproducing kernel Hilbert space (RKHS) $\mathcal{H}$, the resulting $D_{\mathcal{F}}$ is termed the *maximum mean discrepancy* (MMD) (Gretton et al., 2012), and its squared value $\mathrm{MMD}^2(p, q)$ is given as $\|\mu_p - \mu_q\|_{\mathcal{H}}^2$, which equals to:

$$\mathbb{E}_{\boldsymbol{x},\boldsymbol{x}'}[k(\boldsymbol{x}, \boldsymbol{x}')] + \mathbb{E}_{\boldsymbol{y},\boldsymbol{y}'}k(\boldsymbol{y}, \boldsymbol{y}') - 2\mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}[k(\boldsymbol{x}, \boldsymbol{y})] \tag{4}$$

where $\boldsymbol{x}, \boldsymbol{x}' \sim p$, $\boldsymbol{y}, \boldsymbol{y}' \sim q$, and $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is the kernel corresponding to the RKHS satisfying $\mathbb{E}[\sqrt{k(\boldsymbol{x}, \boldsymbol{x})}] < \infty$. If k is a *universal kernel* (Sriperumbudur et al., 2011), which includes the commonly-used Gaussian RBF kernel, then MMD defines a proper metric. In this work we mainly focus on the Euclidean distance kernel defined as

Definition 3 (Euclidean Distance Kernel) *A positive semi-definite kernel function is called Euclidean Distance kernel if it can be represented as:*

$$k(\boldsymbol{x}, \boldsymbol{x}') = f\left(\frac{\|\boldsymbol{x} - \boldsymbol{y}\|_2^2}{\sigma^2}\right),$$

In particular, the commonly-used Gaussian kernels and IMQ kernels are both Euclidean Distance kernels. In practice, σ^2 scales with d for normalization.

Stein's Lemma. When integration under p is difficult, *Stein's method* (Stein et al., 1972) can be used to construct zero-mean test functions w.r.t p. Specifically, for differentiable function f in the Stein Class of p, i.e.,

$$\int_x \nabla_{\boldsymbol{x}}(f(\boldsymbol{x})p(\boldsymbol{x}))d\boldsymbol{x} = 0 \tag{5}$$

The following identity holds:

$$\mathbb{E}_p[\nabla_{\boldsymbol{x}} \log\, p(\boldsymbol{x})f(\boldsymbol{x}) + \nabla_{\boldsymbol{x}} f(\boldsymbol{x})] = \mathbb{E}_p[\mathcal{A}_p f(\boldsymbol{x})] = 0,$$

where $\mathcal{A}_p$ is termed the *Langevin Stein operator* (Gorham and Mackey, 2015), as it arises from applying the *generator method* (Barbour, 1988) to the *overdamped Langevin diffusion*. This identity can be easily verified via integration by parts, given that $(f \cdot p)$ vanishes at boundary. This modified IPM is called the *Stein's discrepancy*:

$$D_{\mathcal{F}}^{\mathcal{A}_p}(p, q) = \sup_{f \in \mathcal{F}} \Big(\mathbb{E}_q[\mathcal{A}_p f(\boldsymbol{x})] - \underbrace{\mathbb{E}_p[\mathcal{A}_p f(\boldsymbol{x})]}_{=0} \Big). \tag{6}$$

Note that the Stein's discrepancy only involves the score of p and thus the normalization constant is not required. When f is restricted in the product RKHS $\mathcal{H}^d$ with inner product $\langle f, g\rangle_{\mathcal{H}^d} = \sum_{i=1}^d \langle f_i, g_i\rangle_{\mathcal{H}}$, the maximum discrepancy, known as *kernel Stein discrepancy* (KSD), can be estimated efficiently from samples (Liu et al., 2016)(Chwialkowski et al., 2016)(Gorham et al., 2016).

Kernel Herding

We now consider the approximation of an intractable distribution $p(\boldsymbol{x})$ with a set of particles $X = \{\boldsymbol{x}_i\}_{i=1}^n$ representing a Dirac mixture. To generate these particles, kernel herding was introduced by (Welling, 2009) for minimizing the MMD between the particles and the target distribution. The herding algorithm proceeds in a greedy manner; Assume the algorithm already selects $\{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_{n-1}\}$, the next particle is chosen based on:

$$\begin{aligned}\boldsymbol{x}_n &\leftarrow \operatorname*{argmin}_x \mathrm{MMD}^2\left(p, \frac{1}{n}\left(\sum_{i=1}^{n-1}\delta_{\boldsymbol{x}_i} + \delta_x\right)\right)\\ &= \operatorname*{argmax}_x \mathbb{E}_{\boldsymbol{y}\sim p}[k(\boldsymbol{x}, \boldsymbol{y})] - \frac{1}{n}\sum_{i=1}^{n-1} k(\boldsymbol{x}, \boldsymbol{x}_i).\end{aligned}$$

Intuitively, the first term encourages sampling in high density areas for the target distribution. The second term discourages sampling at points close to existing samples. It is shown (Welling, 2009; Bach et al., 2012; Huszár and Duvenaud, 2012) that the kernel herding algorithm reduces the MMD at a rate $\mathrm{O}(\frac{1}{N})$, for finite-dimensional Hilbert spaces $\mathcal{H}$.

Stein Variational Gradient Descent

The herding procedure selects particles greedily to minimize its objective MMD^2, adding particles one at a time. One can also jointly optimize all particles to decrease some notion of distance. Let $q_{[\epsilon\Delta]}$ be the distribution of particles after update Δ. SVGD constructs the update direction that maximally decreases the KL divergence:

$$\Delta^* = \operatorname*{argmax}_{\Delta\in\mathcal{H}^d}\left\{-\frac{d}{d\epsilon}D_{KL}(q_{[\epsilon\Delta]}||p)\big|_{\epsilon=0}\right\} = \operatorname*{argmax}_{\Delta\in\mathcal{H}^d}\left\{\mathbb{E}_q[\mathcal{A}_p\Delta(\boldsymbol{x})]\right\}.$$

Constrain Δ in terms of RKHS norm, the update for each particle $\boldsymbol{x}$ can be computed as:

$$\begin{aligned}\Delta^{\mathrm{SVGD}}(\boldsymbol{x}) &= \mathbb{E}_{\boldsymbol{x}'\sim q}[\underbrace{\nabla_{\boldsymbol{x}'}\log p(\boldsymbol{x}')k(\boldsymbol{x}', \boldsymbol{x})}_{\text{driving force}} + \underbrace{\nabla_{\boldsymbol{x}'}k(\boldsymbol{x}', \boldsymbol{x})}_{\text{repulsive force}}]\\ &= \frac{1}{n}\sum_{i=1}^n[\nabla_{\boldsymbol{x}_i}\log p(\boldsymbol{x}_i)k(\boldsymbol{x}_i, \boldsymbol{x}) + \nabla_{\boldsymbol{x}_i}k(\boldsymbol{x}_i, \boldsymbol{x})] := \frac{1}{n}\sum_{i=1}^n S_1(\boldsymbol{x}_i, \boldsymbol{x}) + \frac{1}{n}\sum_{i=1}^n S_2(\boldsymbol{x}_i, \boldsymbol{x}). \quad (7)\end{aligned}$$

Note that this update rule can also be interpreted as a fixed-point iteration on Stein's discrepancy (6). The typically-used kernels include Gaussian kernel $k(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|_2^2}{2\sigma^2})$ (Liu and Wang, 2016; Zhuo et al., 2017) where $\sigma^2 = \mathrm{O}(d)$; Linear kernel $k(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^\top\boldsymbol{x}' + 1$ (Liu and Wang, 2018); Inverse multi-quadratic (IMQ) kernel $k(\boldsymbol{x}, \boldsymbol{x}') = (1 + \|\boldsymbol{x} - \boldsymbol{x}'\|_2^2/(2\sigma^2))^{-1/2}$ (Gorham and Mackey, 2017).

A.2. Related Works

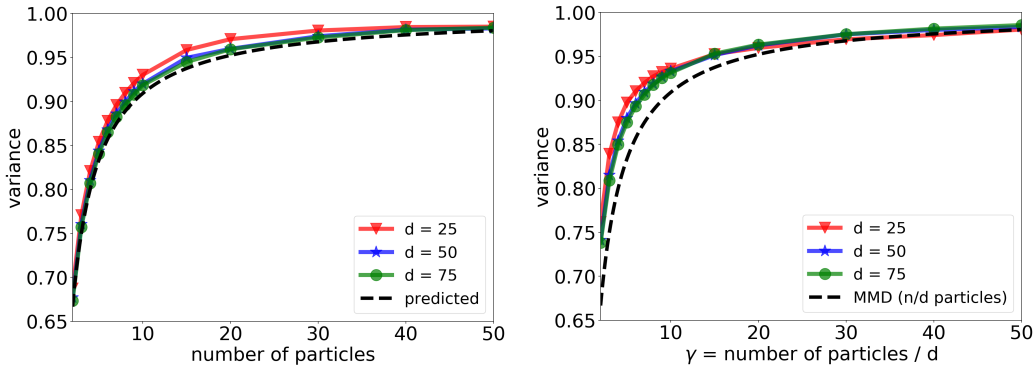
Stein's Lemma. Stein's lemma provides powerful tools in approximating probability distributions and specifying convergence rates (Erdogdu, 2016; Chen et al., 2018). In particular, via Stein's lemma, SVGD (Liu and Wang, 2016) derives an explicit particle updating formula by minimizing the KL divergence with unnormalized targets. With the research of implicit variational inference (Huszár, 2017), Stein's lemma also flourishes score estimation methods (Li and Turner, 2017; Shi et al., 2018) using only random samples from an implicit distribution. Interestingly, Erdogdu et al. (2016) observed that algorithms that are equivalent in expectation via Stein's lemma might have different convergence properties, which aligns with our analysis in Section 3. The "curse of dimensionality" of Stein's lemma-based kernel algorithm has also been studied in Oates et al. (2016).

Guarantees and applications of SVGD. (Liu and Wang, 2018; Lu et al., 2019) characterizes SVGD in the mean-field limit and showed the weak convergence to the target distribution. However, the non-asymptotic convergence is poorly understood and Zhuo et al. (2017); Wang et al. (2018) observed that particles of SVGD tend to underestimate variance in high dimensions, but did not provide a fundamental characterization. Recently, Liu and Wang (2018) shows SVGD using kernels with finite-dimensional feature maps, exactly estimates the expectations for some set of functions, casting SVGD as a moment matching method. On the other hand SVGD has seen fruitful accomplishments across multiple areas. Haarnoja et al. (2017); Liu et al. (2017) adopt SVGD to learn a stochastic sampling network for approximating the optimal policy in Q-learning (Sutton et al., 1998). VGD is also used in meta-learning to quickly obtain parameter samples from training sets (Yoon et al., 2018). Recently works also apply SVGD in terms of Batch Bayesian optimization (Gong et al., 2019) as well as learning diversified mixture models (Wang and Liu, 2019). Leveraging the Markov blanket, SVGD is also applied do inference in graphical models Zhuo et al. (2017); Wang et al. (2018).

A.3. Detail of SVGD with resampling scheme:

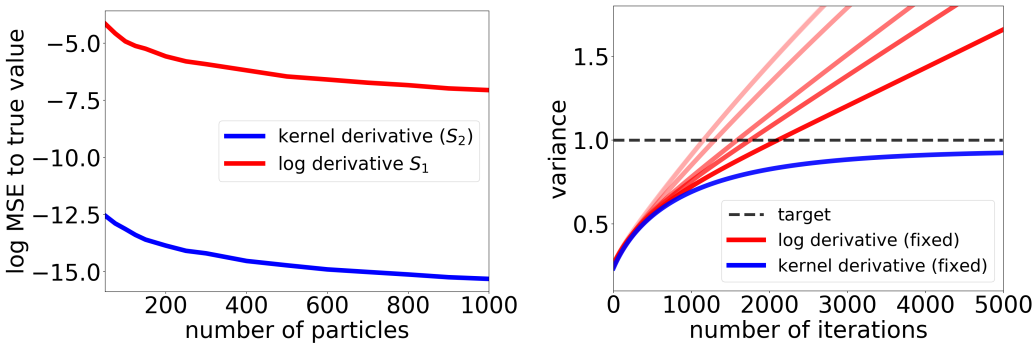
At time T: sample $\{\boldsymbol{x}_i^T\}_{i=0}^n \sim q_0(\boldsymbol{x})$.
for *t=1,2,...,(T-1)* **do**
  $\boldsymbol{x}_i^T \leftarrow \boldsymbol{x}_i^T + \frac{\epsilon}{n}\sum_{j=1}^n S_1(\boldsymbol{x}_j^t, \boldsymbol{x}_i^T) + S_2(\boldsymbol{x}_j^t, \boldsymbol{x}_i^T)$.
end

A.4. Additional Figures



(a) Variance of MMD-descent. (b) Variance of SVGD ($N = nd$).

Figure 3: dimension-averaged marginal variance of particles converged under (a) MMD-descent with n particles, and (b) SVGD with $N = nd$ particles. Note that the equilibrium variance of MMD-descent with n particles follows similar trend as SVGD with nd particles.



(a) MSE in estimating S_1 and S_2 for IMQ kernel. (b) MMD with IMQ kernel.

Figure 4: (a) Integration by parts with the IMQ kernel also leads to a large discrepancy in variance. (b) MMD with IMQ kernel also leads to divergence under Δ_1^{MMD} with fixed target samples, whereas Δ_2^{MMD} with fixed target samples correctly estimates the variance.

A.5. Additional Propositions

Proposition 4 (Fixed-Sample MMD Convergence) *Let $Y = \{\boldsymbol{y}_i\}_{i=1}^m$ be n independent random samples from target distribution p. Assume the kernel is bounded by $0 \leq k(\boldsymbol{z}, \boldsymbol{z}') \leq K$. Let $X^\star = \{\boldsymbol{x}_i\}_{i=1}^n$ be the optimum performing MMD updates based on Δ_2^{MMD} using samples Y. We have*

$$\Pr_Y[\mathrm{MMD}(X^\star, p) \geq \epsilon] \leq \frac{6}{\epsilon}\sqrt{\frac{K}{\min(m,n)}}. \tag{8}$$

Proof:

$$\begin{aligned}
\mathbb{E}_Y[\mathrm{MMD}(X^\star, p)] &= \mathbb{E}_Y[\sup_{f\in\mathcal{H}}(\frac{1}{m}\sum_{i=1}^m f(\boldsymbol{x}_i) - \mathbb{E}_p f(x))] \\
&= \mathbb{E}_Y[\sup_{f\in\mathcal{H}}[(\frac{1}{m}\sum_{i=1}^m f(\boldsymbol{x}_i) - \frac{1}{n}\sum_{i=1}^n f(\boldsymbol{y}_i)) - (\mathbb{E}_p f(x) - \frac{1}{n}\sum_{i=1}^n f(\boldsymbol{y}_i))]] \\
&\leq \mathbb{E}_Y[\sup_{f\in\mathcal{H}}(\frac{1}{m}\sum_{i=1}^m f(\boldsymbol{x}_i) - \frac{1}{n}\sum_{i=1}^n f(\boldsymbol{y}_i)) + \sup_{f\in\mathcal{H}}(\mathbb{E}_p f(x) - \frac{1}{n}\sum_{i=1}^n f(\boldsymbol{y}_i))] \\
&\overset{A}{\leq} \mathbb{E}_{Y,Z\sim p}[\sup_{f\in\mathcal{H}}(\frac{1}{m}\sum_{i=1}^m f(\boldsymbol{z}_i) - \frac{1}{n}\sum_{i=1}^n f(\boldsymbol{y}_i)) + \sup_{f\in\mathcal{H}}(\mathbb{E}_p f(x) - \frac{1}{n}\sum_{i=1}^n f(\boldsymbol{y}_i))] \\
&= \mathbb{E}_{Y,Z\sim p}[\mathrm{MMD}(Z, Y) + \mathrm{MMD}(Y, p)] \\
&\overset{B}{\leq} 2[\sqrt{\frac{K}{m}} + \sqrt{\frac{K}{n}}] + 2\sqrt{\frac{K}{n}} \\
&\leq 6\sqrt{\frac{K}{\min(m,n)}}.
\end{aligned} \tag{9}$$

Where A is because that $X^\star$ attains smallest MMD with Y for all m particles, thus its MMD is no-smaller than m random samples Z from distribution p. B follows from Gretton et al. (2012).

Now based on Markov's Inequality, we have for any $\epsilon > 0$,

$$\Pr_Y[\mathrm{MMD}(X^\star, p) \geq \epsilon] \leq \frac{\mathbb{E}_Y[\mathrm{MMD}(X^\star, p)]}{\epsilon} \leq \frac{6}{\epsilon}\sqrt{\frac{K}{\min(m,n)}}. \tag{10}$$

Appendix B. Proof of Technical Results

B.1. Proof of Proposition 1

Given a Gaussian target $p(\boldsymbol{y}) = \mathcal{N}(0, I_d)$ and Gaussian RBF kernel, the expected value $\mu_{\boldsymbol{x}}$ can be given in closed form as $\mu_{\boldsymbol{x}} = -\frac{\sigma^d}{(1+\sigma^2)^{d/2+1}} \exp\left(-\frac{\|\boldsymbol{x}\|_2^2}{2+2\sigma^2}\right) \boldsymbol{x}$. We compute the mean squared error of the estimates as follow:

$$
\begin{aligned}
\mathrm{MSE}_p[S_1(\boldsymbol{y}, \boldsymbol{x})] &= \int_{\boldsymbol{y}} \|S_1(\boldsymbol{y}, \boldsymbol{x}) - \mu_{\boldsymbol{x}}\|_2^2 \, p(\boldsymbol{y}) d\boldsymbol{y} \\
&= \int_{\boldsymbol{y}} \| - \boldsymbol{y} k(\boldsymbol{x}, \boldsymbol{y}) - \mu_{\boldsymbol{x}}\|_2^2 \, p(\boldsymbol{y}) d\boldsymbol{y} \\
&= \int_{\boldsymbol{y}} k^2(\boldsymbol{x}, \boldsymbol{y}) \boldsymbol{y}^T \boldsymbol{y} \, p(\boldsymbol{y}) d\boldsymbol{y} + 2\mu_x^T \int_{\boldsymbol{y}} k(\boldsymbol{x}, \boldsymbol{y}) \boldsymbol{y} \, p(\boldsymbol{y}) d\boldsymbol{y} + \mu_{\boldsymbol{x}}^T \mu_{\boldsymbol{x}} \int_{\boldsymbol{y}} p(\boldsymbol{y}) d\boldsymbol{y} \\
&= \frac{e^{-\frac{\|\boldsymbol{x}\|_2^2}{2+\sigma^2}} \sigma^d}{(2+\sigma^2)^{d/2+1}} (2\|\boldsymbol{x}\|_2^2 + d\sigma^2) - \frac{2e^{-\frac{\|\boldsymbol{x}\|_2^2}{1+\sigma^2}} \sigma^{2d}}{(1+\sigma^2)^{d+2}} \|\boldsymbol{x}\|_2^2 + \frac{e^{-\frac{\|\boldsymbol{x}\|_2^2}{1+\sigma^2}} \sigma^{2d}}{(1+\sigma^2)^{d+2}} \|\boldsymbol{x}\|_2^2 \\
&= \|\boldsymbol{x}\|_2^2 \Big[\frac{2e^{-\frac{\|\boldsymbol{x}\|_2^2}{2+\sigma^2}}}{\sigma^2+2} \big(\frac{\sigma^2}{2+\sigma^2}\big)^{d/2} - \frac{e^{-\frac{\|\boldsymbol{x}\|_2^2}{1+\sigma^2}}}{(\sigma^2+1)^2} \big(\frac{\sigma^2}{1+\sigma^2}\big)^d \Big] + d e^{-\frac{\|\boldsymbol{x}\|_2^2}{2+\sigma^2}} \big(\frac{\sigma^2}{2+\sigma^2}\big)^{d/2+1}.
\end{aligned}
$$

Similarly for the kernel derivative S_2 we have,

$$
\begin{aligned}
\mathrm{MSE}_p[S_2(\boldsymbol{y}, \boldsymbol{x})] &= \int_{\boldsymbol{y}} \|S_2(\boldsymbol{y}, \boldsymbol{x}) - \mu_{\boldsymbol{x}}\|_2^2 \, p(\boldsymbol{y}) d\boldsymbol{y} \\
&= \int_{\boldsymbol{y}} \| \frac{\boldsymbol{x} - \boldsymbol{y}}{\sigma^2} k(\boldsymbol{x}, \boldsymbol{y}) - \mu_{\boldsymbol{x}}\|_2^2 \, p(\boldsymbol{y}) d\boldsymbol{y} \\
&= \frac{1}{\sigma^4} \int_{\boldsymbol{y}} k^2(\boldsymbol{x}, \boldsymbol{y}) \boldsymbol{y}^T \boldsymbol{y} \, p(\boldsymbol{y}) d\boldsymbol{y} - \frac{2\boldsymbol{x}}{\sigma^4} \int_{\boldsymbol{y}} k^2(\boldsymbol{x}, \boldsymbol{y}) \boldsymbol{y} \, p(\boldsymbol{y}) d\boldsymbol{y} + \frac{2\mu_{\boldsymbol{x}}}{\sigma^2} \int_{\boldsymbol{y}} k(\boldsymbol{x}, \boldsymbol{y}) \boldsymbol{y} \, p(\boldsymbol{y}) d\boldsymbol{y} \\
&\quad + \frac{\boldsymbol{x}^T \boldsymbol{x}}{\sigma^4} \int_{\boldsymbol{y}} k^2(\boldsymbol{x}, \boldsymbol{y}) \, p(\boldsymbol{y}) d\boldsymbol{y} - \frac{2\boldsymbol{x}^T \mu_{\boldsymbol{x}}}{\sigma^2} \int_{\boldsymbol{y}} k(\boldsymbol{x}, \boldsymbol{y}) \, p(\boldsymbol{y}) d\boldsymbol{y} + \mu_{\boldsymbol{x}}^T \mu_{\boldsymbol{x}} \int_{\boldsymbol{y}} p(\boldsymbol{y}) d\boldsymbol{y} \\
&= \frac{e^{-\frac{\|\boldsymbol{x}\|_2^2}{2+\sigma^2}} \sigma^{d-4}}{(2+\sigma^2)^{d/2+1}} (2\|\boldsymbol{x}\|_2^2 + d\sigma^2) - \frac{4e^{-\frac{\|\boldsymbol{x}\|_2^2}{2+\sigma^2}} \sigma^{d-4}}{(2+\sigma^2)^{d/2+1}} \|\boldsymbol{x}\|_2^2 - \frac{2e^{-\frac{\|\boldsymbol{x}\|_2^2}{1+\sigma^2}} \sigma^{2d-2}}{(1+\sigma^2)^{d+2}} \|\boldsymbol{x}\|_2^2 \\
&\quad + \frac{e^{-\frac{\|\boldsymbol{x}\|_2^2}{2+\sigma^2}} \sigma^{d-4}}{(2+\sigma^2)^{d/2+1}} (2+\sigma^2) \|\boldsymbol{x}\|_2^2 + \frac{2e^{-\frac{\|\boldsymbol{x}\|_2^2}{1+\sigma^2}} \sigma^{2d-2}}{(1+\sigma^2)^{d+1}} \|\boldsymbol{x}\|_2^2 + \frac{e^{-\frac{\|\boldsymbol{x}\|_2^2}{1+\sigma^2}} \sigma^{2d}}{(1+\sigma^2)^{d+2}} \|\boldsymbol{x}\|_2^2 \\
&= \frac{e^{-\frac{\|\boldsymbol{x}\|_2^2}{2+2\sigma^2}}}{\sigma^4} \big(\frac{\sigma^2}{2+\sigma^2}\big)^{d/2+1} (d + \|\boldsymbol{x}\|_2^2) + \frac{3e^{-\frac{\|\boldsymbol{x}\|_2^2}{1+\sigma^2}}}{(1+\sigma^2)^2} \big(\frac{\sigma^2}{1+\sigma^2}\big)^d \|\boldsymbol{x}\|_2^2.
\end{aligned}
$$

The simplification above largely follows from $\mathbb{E}_{\boldsymbol{x} \sim \mathcal{N}(\mu, \Sigma)}[\|\boldsymbol{x}\|_2^2] = \mu^T \mu + Tr(\Sigma)$.
Given the bandwidth heuristic $\sigma \in \Theta(\sqrt{d})$ and $\|\boldsymbol{x}\|_2 = d$, one can easily obtain:

$$
\mathrm{MSE}_p[S_2(\boldsymbol{y}, \boldsymbol{x})] \in \Theta(d^{-1}), \quad \mathrm{MSE}_p[S_1(\boldsymbol{y}, \boldsymbol{x})] \in \Theta(d).
$$

Appendix C. Proof of Proposition 4

In this section we aim to calculate the variance of SVGD and MMD-Descent in learning unit Gaussian target under the scaling of $n, d \to \infty$ with $\lim_{d,n\to\infty} n/d = \gamma \in (0, \infty)$. Since both SVGD and MMD-descent form an interacting particle system, one can no longer treat the converged particles as i.i.d. samples from some distribution. We therefore assume the following on the converged fixed point, which essentially entails that the particles spread evenly in the space, have concentrated norm and only correlate weakly.

Assumption A1. Unit Gaussian Target Distribution: $p(\boldsymbol{y}) \propto \exp\left(-\frac{1}{2}\boldsymbol{y}^\top \boldsymbol{y}\right)$; Gaussian RBF Kernel: $k(\boldsymbol{x}, \boldsymbol{y}) = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|_2^2}{2\sigma^2}\right)$.

Assumption A2. At the converged fixed point of SVGD or MMD-descent, there exists $\xi > 0$ such that as $d \to \infty$, $\Pr[\max_i |d^{-1}\|\boldsymbol{x}_i\|_2^2 - v| < \epsilon] \to 1$ and $\Pr[\max_{i,j} |d^{-1}\boldsymbol{x}_i^\top \boldsymbol{x}_j + (n-1)^{-1}v| < \epsilon] \to 1$ holds for $\epsilon = O(d^{-1/2-\xi})$.

Under assumption A1 and A2, we are able to compute the asymptotic variance of both SVGD and MMD-descent. We first calculate the SVGD variance with $d, n \to \infty$, $n/d \to \gamma$.

C.1. Computing the SVGD variance with $d, n \to \infty$

In this subsection we consider the asymptotical scaling of n, d where $n, d \to \infty$ and $n/d \to \gamma$. We solve the stationary point of SVGD update Eq (7), where

$$\Delta(\boldsymbol{x}_k) = \frac{1}{n}\sum_{i=1}^{n}\Big[-k(\boldsymbol{x}_i, \boldsymbol{x}_k)\boldsymbol{x}_i + \frac{1}{dv}k(\boldsymbol{x}_i, \boldsymbol{x}_k)(\boldsymbol{x}_k - \boldsymbol{x}_i)\Big] = \boldsymbol{0} \tag{11}$$

for all k, i.e.

$$\sum_{i=1}^{n} k(\boldsymbol{x}_i, \boldsymbol{x}_k)\boldsymbol{x}_i = \frac{1}{dv+1}\sum_{i=1}^{n} k(\boldsymbol{x}_i, \boldsymbol{x}_k)\cdot \boldsymbol{x}_k. \tag{12}$$

Therefore for Eq (12) we have for LHS

$$\text{LHS} = \sum_{i=1}^{n} k(\boldsymbol{x}_i, \boldsymbol{x}_k)\boldsymbol{x}_i = X\boldsymbol{k}_k, \tag{13}$$

where $X = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{d\times n}$, $\boldsymbol{k}_k = [k(\boldsymbol{x}_1, \boldsymbol{x}_k), \cdots, k(\boldsymbol{x}_n, \boldsymbol{x}_k)]^\top \in \mathbb{R}^n$ (i.e. the k-th column of kernel $K \in \mathbb{R}^{n\times n}$). As for the RHS of Eq (12), note that assumption A2 ensures the following Taylor expansion around its concentrated value for $i \neq k$

$$k(\boldsymbol{x}_i, \boldsymbol{x}_k) = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_k\|_2^2}{2dv}\right) = e^{-1} + O(\epsilon), \tag{14}$$

and for $i = k$ we have $k(\boldsymbol{x}_k, \boldsymbol{x}_k) = 1$. This immediately gives

$$\text{RHS} = \frac{1}{dv+1}\sum_{i=1}^{n} k(\boldsymbol{x}_i, \boldsymbol{x}_k)\cdot \boldsymbol{x}_k = \left(\frac{n+e-1}{(dv+1)e} + O(\epsilon)\right)\boldsymbol{x}_k. \tag{15}$$

Equating the RHS and LHS of Eq (12) in matrix form (over all k) we have

$$X \cdot K = \frac{n+e-1}{(dv+1)e} X + X \cdot \mathrm{diag}(\boldsymbol{\epsilon}) = mX + X \cdot \mathrm{diag}(\boldsymbol{\epsilon}), \tag{16}$$

where $m = (n+e-1)/(dv+1)e$ and $\mathrm{diag}(\boldsymbol{\epsilon})$ is a square matrix where the i-th diagonal is ϵ_i with $\epsilon_i = O(\epsilon)$. Therefore

$$X \cdot (K - mI_n - \mathrm{diag}(\boldsymbol{\epsilon})) = 0. \tag{17}$$

Denote $A = K - mI_n - \mathrm{diag}(\boldsymbol{\epsilon})$, Note that the K is an *Euclidean random kernel matrix* with $K_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_k) = \exp\left(-(2dv)^{-1} \left\|\boldsymbol{x}_i - \boldsymbol{x}_k\right\|_2^2\right)$, from Theorem 4 in (Bordenave et al., 2013) it follows that the empirical spectrum density of A converges weakly to

$$\mu(A) \to \left(1 - \frac{2}{e} - m\right) + \frac{1}{e}\mu\left(\frac{1}{dv} X^\top X\right) + \mu(\mathrm{diag}(\boldsymbol{\epsilon})), \tag{18}$$

where the *empirical spectrum* of a random matrix $A \in \mathbb{R}^{n\times n}$ is defined as $\mu = n^{-1}\sum_{i=1}^n \delta_{\lambda_i(A)}$. Moreover, denote $S = n/(n-1)I_n - 1/(n-1)\mathbf{1}_n\mathbf{1}_n^\top$, then by the *Hoffman-Wielandt inequality* one has

$$W_2\left(\mu\left(\frac{1}{dv}X^\top X\right), \mu(S)\right) \le \sqrt{\frac{1}{n}\mathrm{tr}\left(\frac{1}{dv}X^\top X - S\right)^2} = \sqrt{n^{-1}\cdot nO(\epsilon)^2} = O(\epsilon) \to 0, \tag{19}$$

where $W_2(\cdot,\cdot)$ is the 2-Wasserstein distance. Hence

$$\mu(A) \to \left(1 - \frac{2}{e} - m\right) + \frac{1}{e}\mu\left(\frac{1}{dv}X^\top X\right) + \mu(\mathrm{diag}(\boldsymbol{\epsilon})) \tag{20}$$

$$\to \left(1 - \frac{2}{e} - m\right) + \frac{1}{e}\mu\left(S\right) + \mu(\mathrm{diag}(\boldsymbol{\epsilon})) \tag{21}$$

$$\to 1 - \frac{2}{e} + \frac{n}{e(n-1)} - m. \tag{22}$$

When $\gamma > 1$ i.e. $d > n$, Equation (17) requires $\mu(A) \to 0$, and hence $m \to 1 - e^{-1}$ when $n \to \infty$. This gives

$$v^{SVGD} \to \frac{n}{d(e-1)} = \frac{1}{e-1}\gamma. \tag{23}$$

C.2. Computing the MMD-Descent Variance with $d, n \to \infty$

The stationary point of MMD-descent satisfies for $\forall k$,

$$\Delta\boldsymbol{x}_k = -\frac{\sigma^d}{(1+\sigma^2)^{d/2+1}} e^{-\frac{\|\boldsymbol{x}_k\|_2^2}{2+2\sigma^2}} \boldsymbol{x}_k + \frac{1}{n\sigma^2}\sum_{i\neq k} k(\boldsymbol{x}_k, \boldsymbol{x}_i)(\boldsymbol{x}_k - \boldsymbol{x}_i) = 0, \tag{24}$$

i.e.

$$\sum_{i=1}^n k(\boldsymbol{x}_k, \boldsymbol{x}_i)\boldsymbol{x}_k - \left(\frac{dv}{1+dv}\right)^{d/2+1} e^{-\frac{\|\boldsymbol{x}_k\|_2^2}{2+2dv}} n\boldsymbol{x}_k = \sum_{i=1}^n k(\boldsymbol{x}_k, \boldsymbol{x}_i)\boldsymbol{x}_i.$$

Under assumption A1, similar to the SVGD case, we have the matrix form of the equilibrium particles

$$\left[1 + e^{-1}(n-1) - \left(\frac{dv}{1+dv}\right)^{d/2+1} e^{-\frac{dv}{2+2dv}} n\right] X + X\mathrm{diag}(\boldsymbol{\epsilon}) = XK. \tag{25}$$

with $\epsilon = o(1)$. As $d, n \to \infty$ with $n/d = \gamma$ we have,

$$1 + e^{-1}(n-1) - \left(\frac{dv}{1+dv}\right)^{d/2+1} e^{-\frac{dv}{2+2dv}} n \to 1 - \frac{1}{e}. \tag{26}$$

Note that $\lim_{d\to\infty} \left(dv/(1+dv)\right)^{d/2+1} = e^{-1/(2v)}$. Thus we have

$$v^{MMD} \to 1. \tag{27}$$

C.3. Missing Derivations

Equivalence of Driving Force in MMD-descent :

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{y}\sim p}[\nabla_{\boldsymbol{x}} k(\boldsymbol{x}, \boldsymbol{y})] \\
=& -\int \nabla_{\boldsymbol{y}} k(\boldsymbol{x}, \boldsymbol{y}) p(\boldsymbol{y}) d\boldsymbol{y} \\
=& -\cancel{k(\boldsymbol{x}, \boldsymbol{y}) p(\boldsymbol{y})\big|_{-\infty}^{+\infty}} + \int k(\boldsymbol{x}, \boldsymbol{y}) \nabla_{\boldsymbol{y}} p(\boldsymbol{y}) d\boldsymbol{y} \\
=& \int k(\boldsymbol{x}, \boldsymbol{y}) p(\boldsymbol{y}) \nabla_{\boldsymbol{y}} \log p(\boldsymbol{y}) d\boldsymbol{y} \\
=& \mathbb{E}_{\boldsymbol{y}\sim p}[\nabla_{\boldsymbol{y}} \log p(\boldsymbol{y}) k(\boldsymbol{x}, \boldsymbol{y})].
\end{aligned}$$

The integration by parts and cancellation of $k(\cdot, \boldsymbol{y})p(\boldsymbol{y})\big|_{-\infty}^{+\infty}$ is identical to the derivation of Stein's lemma, under the assumption that $p(\boldsymbol{y})k(\cdot)$ vanishes at the boundary.

Closed-form Driving Force for Gaussian Target :

$$\begin{aligned}
&\int_{\boldsymbol{y}} p(\boldsymbol{y}) k(\boldsymbol{x}, \boldsymbol{y}) \nabla_{\boldsymbol{y}} \log p(\boldsymbol{y}) d\boldsymbol{y} \\
=& \int_{\boldsymbol{y}} e^{-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|_2^2}{2\sigma^2}} (-\boldsymbol{y}) \frac{1}{\sqrt{(2\pi)^d}} e^{-\frac{\boldsymbol{y}^\top \boldsymbol{y}}{2}} d\boldsymbol{y} \\
=& -\frac{1}{\sqrt{(2\pi)^d}} \int_{\boldsymbol{y}} \boldsymbol{y} e^{-\frac{(\frac{1}{\sqrt{1+\sigma^2}}\boldsymbol{x} - \sqrt{1+\sigma^2}\boldsymbol{y})^2}{2\sigma^2}} e^{-\frac{\boldsymbol{x}^\top \boldsymbol{x}}{2+2\sigma^2}} d\boldsymbol{y} \\
=& -\frac{\sigma^d}{(1+\sigma^2)^{d/2+1}} e^{-\frac{\|\boldsymbol{x}\|_2^2}{2+2\sigma^2}} \boldsymbol{x}.
\end{aligned}$$