Stein Variational Gradient Descent Without Gradient

Jun Han [1] **Qiang Liu** [2]

Abstract

Stein variational gradient decent (SVGD) has been shown to be a powerful approximate inference algorithm for complex distributions. However, the standard SVGD requires calculating the gradient of the target density and cannot be applied when the gradient is unavailable. In this work, we develop a gradient-free variant of SVGD (GF-SVGD), which replaces the true gradient with a surrogate gradient, and corrects the induced bias by re-weighting the gradients in a proper form. We show that our GF-SVGD can be viewed as the standard SVGD with a special choice of kernel, and hence directly inherits the theoretical properties of SVGD. We shed insights on the empirical choice of the surrogate gradient and propose an annealed GF-SVGD that leverages the idea of simulated annealing to improve the performance on high dimensional complex distributions. Empirical studies show that our method consistently outperforms a number of recent advanced gradient-free MCMC methods.

1. Introduction

Approximate inference of complex distributions is a longstanding fundamental computational task in machine learning and statistics. Traditional methods are based on either Markov chain Monte Carlo (MCMC) (Neal et al., 2011; Hoffman & Gelman, 2014), or variational inference (VI) (Blei et al., 2017; Zhang et al., 2017), both of which have their own inherent pros and cons: MCMC is theoretically sound and asymptotically consistent, but is often slow to converge in practice; VI is practically faster but has been known to lack theoretical consistency guarantees.

Stein variational gradient descent (SVGD) (Liu & Wang, 2016) is recently developed to integrate the advantages of MCMC and VI. By leveraging a new type of functional gradient descent of KL divergence on the space of distributions, SVGD directly fits a set of non-parametric particles to the distribution of interest, without introducing additional parametric forms. This makes SVGD both theoretically sound (Liu, 2017), practically fast and sample-efficient thanks to its optimization-based formulation.

Unfortunately, gradient information of the target distribution is not always available in practice. In some cases, the distribution of interest is only available as a black-box density function and the gradient cannot be calculated analytically; in other cases, it may be computationally too expensive to calculate the gradient (Beaumont, 2003; Andrieu & Roberts, 2009; Filippone & Girolami, 2014). The goal of this paper is to extend SVGD to gradient-free settings.

Our Results We propose a simple gradient-free variant of SVGD that replaces the true gradient with a surrogate gradient and uses a form of importance weighting to correct the bias introduced by the surrogate gradient. To give a very quick overview of our method, recall that the standard SVGD approximates a given distribution p on $\mathbb{R}^d$ with a set of particles $\{\boldsymbol{x}_i\}_{i=1}^n$ iteratively updated by $\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i + \frac{\epsilon}{n}\Delta\boldsymbol{x}_i$, where

$$\Delta\boldsymbol{x}_i = \sum_{j=1}^n [\nabla \log p(\boldsymbol{x}_j) k(\boldsymbol{x}_j, \boldsymbol{x}_i) + \nabla_{\boldsymbol{x}_j} k(\boldsymbol{x}_j, \boldsymbol{x}_i)], \quad (1)$$

where $k(\boldsymbol{x}, \boldsymbol{x}')$ is any positive definite kernel; the term with gradient $\nabla \log p$ drives the particles to the high probability regions of p, and the term with $\nabla k(\boldsymbol{x}, \boldsymbol{x}')$ acts as a repulsive force to keep the particles away from each other to quantify the uncertainty.

Our gradient-free SVGD is based on the following generalization of the SVGD update rule:

$$\Delta\boldsymbol{x}_i \propto \sum_{j=1}^n w_j \big[\nabla \log \rho(\boldsymbol{x}_j) k(\boldsymbol{x}_j, \boldsymbol{x}_i) + \nabla_{\boldsymbol{x}_j} k(\boldsymbol{x}_j, \boldsymbol{x}_i)\big], \quad (2)$$

which replaces the true gradient $\nabla \log p$ with a surrogate gradient $\nabla \log \rho$ of an arbitrary auxiliary distribution $\rho(\boldsymbol{x})$, and then uses an importance weight $w_j := \rho(\boldsymbol{x}_j)/p(\boldsymbol{x}_j)$ to correct the bias introduced by the surrogate. Perhaps surprisingly, we show that the new update can be derived as a standard SVGD update by using an importance weighted

[1] Computer Science, Dartmouth College [2] Computer Science, The University of Texas at Austin. Correspondence to: Jun <jun.han.gr@dartmouth.edu>, Qiang <lqiang@cs.utexas.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the authors.

kernel $w(\boldsymbol{x})w(\boldsymbol{x}')k(\boldsymbol{x}, \boldsymbol{x}')$, and hence immediately inherits the theoretical proprieties of SVGD; for example, particles updated by (2) can be viewed as a gradient flow of KL divergence similar to the original SVGD (Liu, 2017).

The performance of the gradient-free update (2) critically depends on the choice of ρ. We study this problem empirically and show that it works efficiently, sometimes even better than the original SVGD, if we take ρ to be an *over-dispersed* estimate of p that covers the support of p well. We further improve the algorithm by combining the gradient-free update (2) with simulated annealing, which we show consistently outperforms a number of other advanced gradient-free Monte Carlo methods, including a gradient-free variant of annealed importance sampling (Neal, 2001), kernel adaptive Metropolis-Hastings (KAMH) (Sejdinovic et al., 2014) and kernel Hamiltonian Monte Carlo (KHMC) (Strathmann et al., 2015).

Discussion and Related Works Almost all gradient-free sampling methods employ some auxiliary (or proposal) distributions that are different, but sufficiently close to the target distribution, followed with some mechanisms to correct the bias introduced by the surrogate distribution. There have been a few number of bias-correction mechanisms underlying most of the gradient-free methods, including importance sampling, rejection sampling and the Metropolis-Hastings rejection trick. The state-of-the-art gradient-free sampling methods are often based on adaptive improvement of the proposals when using these tricks, this includes adaptive importance sampling and rejection sampling (Gilks & Wild, 1992; Cappé et al., 2008; Cotter et al., 2015; Han & Liu, 2017), and adaptive MCMC methods (e.g., Sejdinovic et al., 2014; Strathmann et al., 2015).

Our method is significantly different from these gradient-free sampling algorithms aforementioned in principle, with a number of key advantages. Instead of correcting the bias by either re-weighting or rejecting the samples from the proposal, which unavoidably reduces the effective number of usable particles, our method re-weights the SVGD gradient and steers the update direction of the particles in a way that compensates the discrepancy between the target and surrogate distribution, without directly reducing the effective number of usable particles.

In addition, while the traditional importance sampling and rejection sampling methods require the proposals to be simple enough to draw samples from, our update does not require to draw samples from the surrogate ρ. We can set ρ to be arbitrarily complex as long as we can calculate $\rho(\boldsymbol{x})$ and its gradient. In fact, $\rho(\boldsymbol{x})$ does not even have to be a normalized probability, sidestepping the difficult problem of calculating the normalization constant.

Outline The paper is organized as follows. Section 2 introduces the background of SVGD. Section 3 introduces our novel gradient-free SVGD algorithm, and Section 4 further proposes annealed gradient-free SVGD that combines the advantage of simulated annealing to achieve better performance for complex distributions. We present empirical results in Section 5 and conclude the paper in Section 6.

2. Stein Variational Gradient Descent

Stein variational gradient descent (SVGD) (Liu & Wang, 2016) is a nonparametric variational inference algorithm that iteratively transports a set of particles to approximate a given target distribution by performing a type of functional gradient descent on the KL divergence. We give a quick overview of its main idea in this section.

Let $p(\boldsymbol{x})$ be a positive density function on $\mathbb{R}^d$ which we want to approximate with a set of particles $\{\boldsymbol{x}_i\}_{i=1}^n$. SVGD starts with a set of initial particles $\{\boldsymbol{x}_i\}_{i=1}^n$, and updates the particles iteratively by

$$\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i + \epsilon \boldsymbol{\phi}(\boldsymbol{x}_i), \quad \forall i = 1, \ldots, n, \tag{3}$$

where ϵ is a step size, and $\boldsymbol{\phi}\colon \mathbb{R}^d \to \mathbb{R}^d$ is a velocity field which should be chosen to drive the particle distribution closer to the target. Assume the distribution of the particles at the current iteration is q, and $q_{[\epsilon\boldsymbol{\phi}]}$ is the distribution of the updated particles $\boldsymbol{x}' = \boldsymbol{x} + \epsilon\boldsymbol{\phi}(\boldsymbol{x})$. The optimal choice of $\boldsymbol{\phi}$ can be framed into the following optimization problem:

$$\boldsymbol{\phi}^* = \operatorname*{arg\,max}_{\boldsymbol{\phi}\in\mathcal{F}} \left\{ -\frac{d}{d\epsilon}\mathrm{KL}(q_{[\epsilon\boldsymbol{\phi}]} \,||\, p)\big|_{\epsilon=0} \right\}, \tag{4}$$

where $\mathcal{F}$ is the set of candidate velocity fields, and $\boldsymbol{\phi}$ is chosen in $\mathcal{F}$ to maximize the decreasing rate on the KL divergence between the particle distribution and the target.

In SVGD, $\mathcal{F}$ is chosen to be the unit ball of a vector-valued reproducing kernel Hilbert space (RKHS) $\mathcal{H} = \mathcal{H}_0 \times \cdots \times \mathcal{H}_0$, where $\mathcal{H}_0$ is a RKHS formed by scalar-valued functions associated with a positive definite kernel $k(\boldsymbol{x}, \boldsymbol{x}')$, that is, $\mathcal{F} = \{\boldsymbol{\phi} \in \mathcal{H}\colon ||\boldsymbol{\phi}||_{\mathcal{H}} \le 1\}$. This choice of $\mathcal{F}$ allows us to consider velocity fields in infinite dimensional function spaces while still obtaining computationally tractable solution.

A key step towards solving (4) is to observe that the objective function in (4) is a simple linear functional of $\boldsymbol{\phi}$ that connects to Stein operator (Oates et al., 2017; Gorham & Mackey, 2015; Liu & Wang, 2016; Gorham & Mackey, 2017; Chen et al., 2018),

$$-\frac{d}{d\epsilon}\mathrm{KL}(q_{[\epsilon\boldsymbol{\phi}]} \,||\, p)\big|_{\epsilon=0} = \mathbb{E}_{x\sim q}[\mathcal{A}_p^\top \boldsymbol{\phi}(\boldsymbol{x})], \tag{5}$$

$$\text{with} \quad \mathcal{A}_p^\top \boldsymbol{\phi}(\boldsymbol{x}) = \nabla_{\boldsymbol{x}} \log p(\boldsymbol{x})^\top \boldsymbol{\phi}(\boldsymbol{x}) + \nabla_{\boldsymbol{x}}^\top \boldsymbol{\phi}(\boldsymbol{x}), \tag{6}$$

where $\mathcal{A}_p$ is a linear operator called *Stein operator* and is formally viewed as a column vector similar to the gradient operator $\nabla_{\boldsymbol{x}}$. The Stein operator $\mathcal{A}_p$ is connected to Stein's identity which shows that the RHS of (5) is zero if $p = q$:

$$\mathbb{E}_{\boldsymbol{x}\sim p}[\mathcal{A}_p^\top \boldsymbol{\phi}(\boldsymbol{x})] = 0. \tag{7}$$

This corresponds to $\frac{d}{d\epsilon}\mathrm{KL}(q_{[\epsilon\boldsymbol{\phi}]} \,||\, p)\big|_{\epsilon=0} = 0$ since there is no way to further decease the KL divergence when $p = q$. Eq. (7) is a simple result of integration by parts assuming the value of $p(\boldsymbol{x})\boldsymbol{\phi}(\boldsymbol{x})$ vanishes on the boundary of the integration domain.

Therefore, the optimization in (4) reduces to

$$\mathbb{D}_{\mathcal{F}}(q||p) \overset{def}{=} \max_{\boldsymbol{\phi}\in\mathcal{F}} \left\{\mathbb{E}_{\boldsymbol{x}\sim q}\left[\mathcal{A}_p^\top \boldsymbol{\phi}(\boldsymbol{x})\right]\right\}, \tag{8}$$

where $\mathbb{D}_{\mathcal{F}}(q \,||\, p)$ is the kernelized Stein discrepancy (KSD) defined in Liu et al. (2016); Chwialkowski et al. (2016).

Observing that (8) is "simple" in that it is a linear functional optimization on a unit ball of a Hilbert space, Liu & Wang (2016) showed that (4) has a simple closed-form solution:

$$\boldsymbol{\phi}^*(\boldsymbol{x}') \propto \mathbb{E}_{\boldsymbol{x}\sim q}[\mathcal{A}_p k(\boldsymbol{x}, \boldsymbol{x}')], \tag{9}$$

where $\mathcal{A}_p$ is applied to variable $\boldsymbol{x}$, and

$$\mathbb{D}^2_{\mathcal{F}}(q \,||\, p) = \mathbb{E}_{\boldsymbol{x},\boldsymbol{x}'\sim q}[\kappa_p(\boldsymbol{x}, \boldsymbol{x}')], \tag{10}$$

where $\kappa_p(\boldsymbol{x}, \boldsymbol{x}') := (\mathcal{A}_p')^\top(\mathcal{A}_p k(\boldsymbol{x}, \boldsymbol{x}'))$ and $\mathcal{A}_p'$ denotes the Stein operator applied on variable $\boldsymbol{x}'$. Here $\kappa_p(\boldsymbol{x}, \boldsymbol{x}')$ can be calculated explicitly in Theorem 3.6 of Liu et al. (2016).

The Stein variational gradient direction $\boldsymbol{\phi}^*$ provides a theoretically optimal direction that drives the particles towards the target p as fast as possible. In practice, SVGD approximates q using a set of particles, yielding update rule (1).

3. Gradient-Free SVGD

The standard SVGD requires the gradient of the target p and cannot be applied when the gradient is unavailable. In this section, we propose a gradient-free variant of SVGD which replaces the true gradient with a surrogate gradient and corrects the bias introduced using an importance weight. We start with introducing a gradient-free variant of Stein's identity and Stein discrepancy.

Gradient-Free Stein's Identity and Stein Discrepancy We can generalize Stein's identity to make it depend on a surrogate gradient $\nabla_{\boldsymbol{x}} \log \rho$ of an arbitrary auxiliary distribution ρ, instead of the true gradient $\nabla_{\boldsymbol{x}} \log p$. The idea is to use importance weights to transform Stein's identity of ρ into an identity regarding p: recall the Stein's identity of ρ:

$$\mathbb{E}_{\boldsymbol{x}\sim \rho}[\mathcal{A}_\rho^\top \boldsymbol{\phi}(\boldsymbol{x})] = 0.$$

It can be easily seen that it is equivalent to the following *importance weighted Stein's identity*:

$$\mathbb{E}_{\boldsymbol{x}\sim p}\left[\frac{\rho(\boldsymbol{x})}{p(\boldsymbol{x})}\mathcal{A}_\rho^\top \boldsymbol{\phi}(\boldsymbol{x})\right] = 0, \tag{11}$$

which is already gradient free since it depends on p only through the value of $p(\boldsymbol{x})$, not the gradient. (11) holds for an arbitrary auxiliary distribution ρ which satisfies $\rho(\boldsymbol{x})/p(\boldsymbol{x}) < \infty$ for any $\boldsymbol{x}$.

Based on identity (11), it is straightforward to define an importance weighted Stein discrepancy

$$\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p) = \max_{\boldsymbol{\phi}\in\mathcal{F}} \left\{\mathbb{E}_{x\sim q}\left[\frac{\rho(\boldsymbol{x})}{p(\boldsymbol{x})}\mathcal{A}_\rho^\top \boldsymbol{\phi}(\boldsymbol{x})\right]\right\}, \tag{12}$$

which is gradient-free if ρ does not depend on the gradient of p. Obviously, this includes the standard Stein discrepancy in Section 2 as special cases: if $\rho = p$, then $\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p) = \mathbb{D}_{\mathcal{F}}(q \,||\, p)$, reducing to the original definition in (8), while if $\rho = q$, then $\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p) = \mathbb{D}_{\mathcal{F}}(p \,||\, q)$, which switches the order of p and q.

It may appear that $\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p)$ strictly generalizes the definition (8) of Stein discrepancy. One of our key observations, however, shows that this is not the case. Instead, $\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p)$ can also be viewed as a special case of $\mathbb{D}_{\mathcal{F}}(q \,||\, p)$, by replacing $\mathcal{F}$ in (8) with

$$w\mathcal{F} := \{w(\boldsymbol{x})\boldsymbol{\phi}(\boldsymbol{x}) \colon \boldsymbol{\phi} \in \mathcal{F}\},$$

where $w(\boldsymbol{x}) = \rho(\boldsymbol{x})/p(\boldsymbol{x})$.

Theorem 1. *Let p, ρ be positive differentiable densities and $w(\boldsymbol{x}) = \rho(\boldsymbol{x})/p(\boldsymbol{x})$. We have*

$$w(\boldsymbol{x})\mathcal{A}_\rho^\top \boldsymbol{\phi}(\boldsymbol{x}) = \mathcal{A}_p^\top\big(w(\boldsymbol{x})\boldsymbol{\phi}(\boldsymbol{x})\big). \tag{13}$$

Therefore, $\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p)$ in (12) is equivalent to

$$\begin{aligned}\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p) &= \max_{\boldsymbol{\phi}\in\mathcal{F}} \left\{\mathbb{E}_{\boldsymbol{x}\sim q}[\mathcal{A}_p^\top\big(w(\boldsymbol{x})\boldsymbol{\phi}(\boldsymbol{x})\big)]\right\} && (14)\\ &= \max_{\boldsymbol{\phi}\in w\mathcal{F}} \left\{\mathbb{E}_{\boldsymbol{x}\sim q}[\mathcal{A}_p^\top \boldsymbol{\phi}(\boldsymbol{x})]\right\} && (15)\\ &= \mathbb{D}_{w\mathcal{F}}(q \,||\, p).\end{aligned}$$

Proof: The proof can be found in the appendix A. □

Identity (13) is interesting because it is *gradient-free* (in terms of $\nabla_{\boldsymbol{x}} \log p$) from the left hand side, but *gradient-dependent* from the right hand side; this is because the $\nabla_{\boldsymbol{x}} \log p$ term in $\mathcal{A}_p$ is cancelled out when applying $\mathcal{A}_p$ on the density ratio $w(\boldsymbol{x}) = \rho(\boldsymbol{x})/p(\boldsymbol{x})$.

It is possible to possible to further extend our method to take $\rho(\boldsymbol{x})$ and $w(\boldsymbol{x})$ to be general *matrix-valued* functions, in which case the operator $\mathcal{A}_p^\top(w(\boldsymbol{x})\phi(\boldsymbol{x}))$ is called diffusion Stein operator in Gorham et al. (2016), corresponding to various forms of Langevin diffusion when taking special values of $w(\boldsymbol{x})$. We leave it as future work to explore ρ.

Algorithm 1 Gradient-Free SVGD (GF-SVGD)

```
Input: Target distribution $p(\boldsymbol{x})$; Surrogate $\rho(\boldsymbol{x})$ and its score function $\boldsymbol{s}_\rho(\boldsymbol{x}) := \nabla_{\boldsymbol{x}} \log \rho$.
Goal: Find particles $\{\boldsymbol{x}_i\}_{i=1}^n$ to approximate $p$.
Initialize particles $\{\boldsymbol{x}_i^0\}_{i=1}^n$ from any distribution $q$.
for iteration $t$ do
    $\boldsymbol{x}_i^{t+1} \leftarrow \boldsymbol{x}_i^t + \Delta\boldsymbol{x}_i^t, \quad \forall i = 1, \ldots, n,$ where
    $\Delta\boldsymbol{x}_i^t = \frac{\epsilon_{t,i}}{Z_t} \sum_{j=1}^n w(\boldsymbol{x}_j^t)\big[\boldsymbol{s}_\rho(\boldsymbol{x}_j^t) k(\boldsymbol{x}_j^t, \boldsymbol{x}_i^t) + \nabla_{\boldsymbol{x}_j} k(\boldsymbol{x}_j^t, \boldsymbol{x}_i^t)\big],$
    where $w(\boldsymbol{x}) := \rho(\boldsymbol{x})/p(\boldsymbol{x})$, $Z_t = \sum_{j=1}^n w(\boldsymbol{x}_j^t)$, and $\epsilon_{t,i}$ is a step size.
end for
```

Gradient-Free SVGD Theorem 1 suggests that by simply multiplying $\boldsymbol{\phi}$ with an importance weight $w(\boldsymbol{x})$ (or replacing $\mathcal{F}$ with $w\mathcal{F}$), one can transform Stein operator $\mathcal{A}_p$ to operator $\mathcal{A}_\rho$, which depends on $\nabla \log \rho$ instead of $\nabla \log p$ (*gradient-free*). This idea can be directly applied to derive a gradient-free extension of SVGD, by updating the particles using velocity fields of form $w(\boldsymbol{x})\boldsymbol{\phi}(\boldsymbol{x})$ from space $w\mathcal{F}$:

$$\boldsymbol{x} \leftarrow \boldsymbol{x} + \epsilon w(\boldsymbol{x})\boldsymbol{\phi}^*(\boldsymbol{x}), \tag{16}$$

where $\boldsymbol{\phi}^*$ maximzies the decrease rate of KL divergence,

$$\boldsymbol{\phi}^* = \arg\max_{\boldsymbol{\phi}\in\mathcal{H}} \big\{\mathbb{E}_q[\mathcal{A}_p^\top (w(\boldsymbol{x})\boldsymbol{\phi}(\boldsymbol{x}))], \text{s.t. } ||\boldsymbol{\phi}||_{\mathcal{H}} \le 1\big\}. \tag{17}$$

Similar to (8), we can derive a closed-form solution for (17) when $\mathcal{H}$ is RKHS. To do this, it is sufficient to recall that if $\mathcal{H}$ is an RKHS with kernel $k(\boldsymbol{x}, \boldsymbol{x}')$, then $w\mathcal{H}$ is also an RKHS, with an "importance weighted kernel" (Berlinet & Thomas-Agnan, 2011)

$$\tilde{k}(\boldsymbol{x}, \boldsymbol{x}') = w(\boldsymbol{x})w(\boldsymbol{x}')k(\boldsymbol{x}, \boldsymbol{x}'). \tag{18}$$

Theorem 2. *When $\mathcal{H}$ is an RKHS with kernel $k(\boldsymbol{x}, \boldsymbol{x}')$, the optimal solution of (14) is $\boldsymbol{\phi}^*/||\boldsymbol{\phi}^*||_{\mathcal{H}}$, where*

$$\begin{aligned} \boldsymbol{\phi}^*(\cdot) &= \mathbb{E}_{\boldsymbol{x}\sim q}[\mathcal{A}_p(w(\boldsymbol{x})k(\boldsymbol{x}, \cdot))] && (19) \\ &= \mathbb{E}_{\boldsymbol{x}\sim q}[w(\boldsymbol{x})\mathcal{A}_\rho k(\boldsymbol{x}, \cdot)], && (20) \end{aligned}$$

where the Stein operator $\mathcal{A}_\rho$ is applied to variable $\boldsymbol{x}$, $\mathcal{A}_\rho k(\boldsymbol{x}, \cdot) = \nabla_{\boldsymbol{x}} \log \rho(\boldsymbol{x}) k(\boldsymbol{x}, \cdot) + \nabla_{\boldsymbol{x}} k(\boldsymbol{x}, \cdot)$. Correspondingly, the optimal decrease rate of KL divergence in (17) equals the square of $\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p)$, which equals

$$\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p) = (\mathbb{E}_{\boldsymbol{x},\boldsymbol{x}'\sim q}[w(\boldsymbol{x})w(\boldsymbol{x}')\kappa_\rho(\boldsymbol{x}, \boldsymbol{x}')])^{\frac{1}{2}}, \tag{21}$$

where $\kappa_\rho(\boldsymbol{x}, \boldsymbol{x}') = (\mathcal{A}'_\rho)^\top(\mathcal{A}_\rho k(\boldsymbol{x}, \boldsymbol{x}'))$ and $\mathcal{A}'_\rho$ is the Stein operator applied on variable $\boldsymbol{x}'$.

Proof: The proof can be found in the appendix B. □

The form in (21) allows us to estimate $\mathbb{D}_{\mathcal{F},\rho}(q \,||\, p)$ empirically either using U-statistics or V-statistics when q is observed through an i.i.d. sample, with the advantage of being gradient-free. Therefore, it can be directly applied to construct gradient-free methods for goodness-of-fit tests (Liu et al., 2016; Chwialkowski et al., 2016) and black-box importance sampling (Liu & Lee, 2017) when the gradient of p is unavailable. We leave this to future works.

Using the gradient-free form of $\boldsymbol{\phi}^*$ in (20), we can readily derive a gradient-free SVGD update $\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i + \Delta\boldsymbol{x}_i$, with

$$\Delta\boldsymbol{x}_i = \frac{\epsilon_i}{Z} \sum_{j=1}^n w(\boldsymbol{x}_j)\mathcal{A}_\rho k(\boldsymbol{x}_j, \boldsymbol{x}_i), \tag{22}$$

where the operator $\mathcal{A}_\rho$ is applied on variable $\boldsymbol{x}_j$, and we set $Z = n$, viewed as a normalization constant, and $\epsilon_i = \epsilon w(\boldsymbol{x}_i)$, viewed as the step size of particle $\boldsymbol{x}_i$.

Since ϵ_i/Z is a scalar, we can change it in practice without altering the set of fixed points of the update. In practice, because the variability of the importance weight $w(\boldsymbol{x}_i)$ can be very large, making the updating speed of different particles significantly different, we find it is empirically better to determine ϵ_i directly using off-the-shelf step size schemes such as Adam (Kingma & Ba, 2015).

In practice, we also replace $Z = n$ with a self-normalization factor $Z = \sum_{j=1}^n w(\boldsymbol{x}_j)$ (see Algorithm 1). We find this makes tuning step sizes become easier in practice, and more importantly, avoids to calculate the normalization constant of either $p(\boldsymbol{x})$ or $\rho(\boldsymbol{x})$. This sidesteps the critically challenging problem of calculating the normalization constant and allows us to essentially choose $\rho(\boldsymbol{x})$ to be an arbitrary positive differentiable function once we can calculate its value and gradient.

Choice of the Auxiliary Distribution $\rho(\boldsymbol{x})$ Obviously, the performance of gradient-free SVGD (GF-SVGD) critically depends on the choice of the auxiliary distribution $\rho(\boldsymbol{x})$. Theoretically, gradient-free SVGD is just a standard SVGD with the importance weighted kernel $\tilde{k}(\boldsymbol{x}, \boldsymbol{x}')$. Therefore, the optimal choice of ρ is essentially the problem of choosing an optimal kernel for SVGD, which, unfortunately, is a difficult, unsolved problem.

In this work, we take a simple heuristic that sets $\rho(\boldsymbol{x})$ to approximate $p(\boldsymbol{x})$. This is based on the justification that if the original $k(\boldsymbol{x}, \boldsymbol{x}')$ has been chosen to be optimal or "reasonably well", we should take $\rho(\boldsymbol{x}) \approx p(\boldsymbol{x})$ so that $\tilde{k}(\boldsymbol{x}, \boldsymbol{x}')$ is close to $k(\boldsymbol{x}, \boldsymbol{x}')$ and GF-SVGD will have similar performance as the original SVGD. In this way, the problem of choosing the optimal auxiliary distribution $\rho(\boldsymbol{x})$ and the optimal kernel $k(\boldsymbol{x}, \boldsymbol{x}')$ is separated, and different kernel selection methods can be directly plugged into the algorithm. In practice, we find that $\rho(\boldsymbol{x}) \approx p(\boldsymbol{x})$ serves a reasonable

Algorithm 2 Annealed SVGD (A-SVGD)

```
Inputs: $p(\boldsymbol{x})$, distribution path $\{p_t\}_{t=1}^T$ with $p_T = p$.
Initialize particles $\{\boldsymbol{x}_i^0\}_{i=1}^n$ from any distribution.
for iteration $t = 0, \cdots, T-1$ do
    Update the particles to get $\{\boldsymbol{x}_i^{t+1}\}_{i=1}^n$ by running the typical SVGD with $p_{t+1}$ as the target for $m$ steps.
end for
Output: $\{\boldsymbol{x}_i^T\}_{i=1}^n$ as an approximation of $p$.
Remark: $m = 1$ is sufficient when $T$ is large.
```

heuristic when using Gaussian RBF kernel $k(\boldsymbol{x}, \boldsymbol{x}')$. Interestingly, our empirical observation shows that a widely spread $\rho(\boldsymbol{x})$ tends to give better and more stable results than peaky $\rho(\boldsymbol{x})$. In particular, Figure 1 in the experiment section shows that in the case when both $p(\boldsymbol{x})$ and $\rho(\boldsymbol{x})$ are Gaussian and RBF kernel is used, the best performance is achieved when the variance of $\rho(\boldsymbol{x})$ is larger than the variance of p. In fact, the gradient-free SVGD update (22) still makes sense even when $\rho(\boldsymbol{x}) = 1$, corresponding to an improper distribution with infinite variance:

$$\Delta \boldsymbol{x}_i = \frac{\epsilon_i}{Z} \sum_{j=1}^n \frac{1}{p(\boldsymbol{x}_j)} \nabla_{\boldsymbol{x}_j} k(\boldsymbol{x}_j, \boldsymbol{x}_i). \qquad (23)$$

This update is interestingly simple; it has only a repulsive force and relies on an inverse probability $1/p(\boldsymbol{x})$ to adjust the particles towards the target $p(\boldsymbol{x})$. We should observe that it is *as general as* the GF-SVGD update (22) (and hence the standard SVGD update (1)), because if we replace $k(\boldsymbol{x}, \boldsymbol{x}')$ with $\rho(\boldsymbol{x})\rho(\boldsymbol{x}')k(\boldsymbol{x}, \boldsymbol{x}')$, (23) reduces back to (22). All it matters is the choice of the kernel function. With a "typical" kernel such as RBF kernel, we empirically find that the particles by the update (23) can estimate the mean parameter reasonably well (although not optimally), but tend to overestimate the variance because the repulsive force dominates; see Figure 1.

4. Annealed Gradient-Free SVGD

In practice, it may be difficult to directly find $\rho(\boldsymbol{x})$ that closely approximates the target p, causing the importance weights to have undesirably large variance and deteriorate the performance. In this section, we introduce an annealed GF-SVGD algorithm that overcomes the difficulty of choosing ρ and improves the performance by iteratively approximating a sequence of distributions which interpolate the target distribution with a simple initial distribution. In the sequel, we first introduce the annealed version of the basic SVGD and then its combination with GF-SVGD.

Annealed SVGD (A-SVGD) is a simple combination of SVGD and simulated annealing, and has been discussed by Liu et al. (2017) in the setting of reinforcement learning. Let $p_0(\boldsymbol{x})$ be a simple initial distribution. We define a path of distributions that interpolate between $p_0(\boldsymbol{x})$ and $p(\boldsymbol{x})$:

$$p_t(\boldsymbol{x}) \propto p_0(\boldsymbol{x})^{1-\alpha_t} p(\boldsymbol{x})^{\alpha_t},$$

where $0 = \alpha_0 < \alpha_1 < \cdots < \alpha_T = 1$ is a set of temperatures. Annealed SVGD starts from a set of particle $\{x_i^0\}_{i=1}^n$ drawn from p_0, and at the t-th iteration, updates the particles so that $\{x_i^{t+1}\}_{i=1}^n$ approximates the intermediate distribution p_{t+1} by running m steps of SVGD with p_{t+1} as the target. See Algorithm 2. In practice, $m = 1$ is sufficient when T is large. It is useful to consider the special case when $p_0 = const$, and hence $p_t(\boldsymbol{x}) \propto p(\boldsymbol{x})^{\alpha_t}$, yielding an annealed SVGD update of form

$$\Delta \boldsymbol{x}_i = \frac{\epsilon}{n} \sum_{j=1}^n [\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}_j) k(\boldsymbol{x}_j, \boldsymbol{x}_i) + \frac{1}{\alpha_t} \nabla_{\boldsymbol{x}_j} k(\boldsymbol{x}_j, \boldsymbol{x}_i)],$$

where the repulsive force is weighted by the inverse temperature $1/\alpha_t$. As α_t increases from 0 to 1, the algorithm starts with a large repulsive force and gradually decreases it to match the temperature of the distribution of interest. This procedure is similar to the typical simulated annealing, but enforces the diversity of the particles using the deterministic repulsive force, instead of random noise.

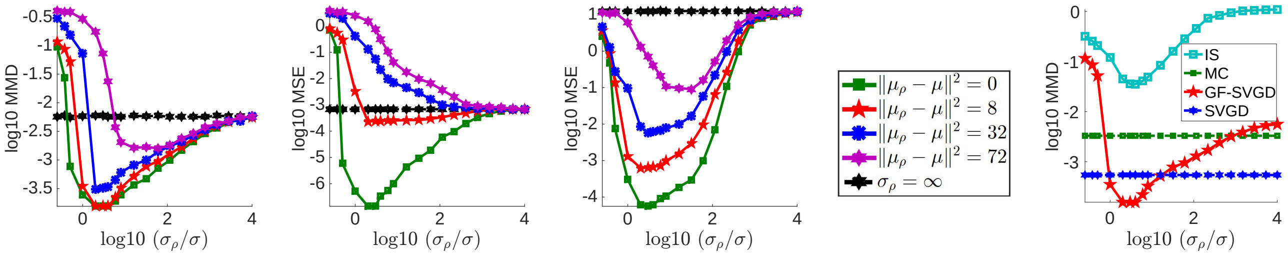
Algorithm 3 Annealed GF-SVGD (AGF-SVGD)

```
Input: Target distribution $p(\boldsymbol{x})$; initial distribution $p_0(\boldsymbol{x})$; intermediate distributions $\{p_t\}_{t=1}^T$.
Goal: Particles $\{\boldsymbol{x}_i\}_{i=1}^n$ to approximate $p(\boldsymbol{x})$.
Initialize particles $\{\boldsymbol{x}_i^0\}_{i=1}^n$ drawn from $p_0$.
for iteration $t = 0, \cdots, T-1$ do
    $$\boldsymbol{x}_i^{t+1} \leftarrow \boldsymbol{x}_i^t + \Delta \boldsymbol{x}_i^t, \ \ \forall i = 1, \ldots, n, \ \text{ where}$$
    $$\Delta \boldsymbol{x}_i^t = \frac{\epsilon_{t,i}}{Z_t} \sum_{j=1}^n w_j^t \big[\boldsymbol{s}_{j,t+1}^\rho k(\boldsymbol{x}_j^t, \boldsymbol{x}_i^t) + \nabla_{\boldsymbol{x}_j} k(\boldsymbol{x}_j^t, \boldsymbol{x}_i^t)\big],$$
    where $\boldsymbol{s}_{j,t+1}^\rho = \nabla_{\boldsymbol{x}} \log \rho_{t+1}(\boldsymbol{x}_j^t)$ and $\rho_{t+1}$ is defined in (24); $w_j^t = \rho_{t+1}(\boldsymbol{x}_j^t)/p_{t+1}(\boldsymbol{x}_j^t)$, $Z_t = \sum_{j=1}^n w_j^t$.
end for
Output: $\{\boldsymbol{x}_i^T\}_{i=1}^n$ to approximate $p$.
```

Annealed Gradient-Free SVGD (AGF-SVGD) is the gradient-free version of annealed SVGD which replaces the SVGD update with an GF-SVGD update. Specifically, at the t-th iteration when we want to update the particles to match p_{t+1}, we use a GF-SVGD update with auxiliary distribution $\rho_{t+1} \approx p_{t+1}$, which we construct by using a simple kernel curve estimation

$$\rho_{t+1}(\boldsymbol{x}) \propto \sum_{j=1}^n p_{t+1}(\boldsymbol{x}_j^t) k_\rho(\boldsymbol{x}_j^t, \boldsymbol{x}), \qquad (24)$$

where k_ρ is a smoothing kernel (which does not have to be positive definite). Although there are other ways to approximate p_{t+1}, this simple heuristic is computationally



(a) MMD (b) Mean (c) Variance (d) MMD

Figure 1. Results of GF-SVGD on 2D multivariate Gaussian distribution as we change the mean $\boldsymbol{\mu}_\rho$ and variance σ_ρ of the surrogate $\rho(\boldsymbol{x})$. We can see that the best performance is achieved by matching the mean of ρ and the target p ($\boldsymbol{\mu}_\rho = \boldsymbol{\mu}$), and making σ_ρ slightly larger than the variance σ of p (e.g., $\log 10(\sigma_\rho/\sigma) \approx 0.5$ or $\sigma_\rho \approx 3\sigma$). (d) uses the same setting with $||\boldsymbol{\mu}_\rho - \boldsymbol{\mu}||^2 = 8$, but also adds the result of exact Monte Carlo sampling, gradient-based SVGD, and importance sampling (IS) whose proposal is ρ, the same as the auxiliary distribution used by GF-SVGD shown in the red curve. We use $n = 100$ particles in this plot.

fast, and the usage of smoothing kernel makes ρ_{t+1} an *over-dispersed* estimate which we show perform well in practice (see Figure 1). Note that here ρ_{t+1} is constructed to *fit smooth curve* p_{t+1}, which leverages the function values of $p_{t+1}(\boldsymbol{x})$ and is insensitive to the actual distribution of the current particles $\{\boldsymbol{x}_j^t\}$. It would be less robust to construct ρ_{t+1} *as a density estimator of distribution* p_{t+1} because the actual distribution of the particles may deviate from what we expect in practice.

The procedure is organized in Algorithm 3. Combining the idea of simulated annealing with gradient-free SVGD makes it easier to construct an initial surrogate distribution and estimate a good auxiliary distribution at each iteration, decreasing the variance of the importance weights. We find that it significantly improves the performance over the basic GF-SVGD for complex target distributions.

5. Empirical Results

We test our proposed algorithms on both synthetic and real-world examples. We start with testing our methods on simple multivariate Gaussian and Gaussian mixture models, developing insights on the optimal choice of the auxiliary distribution. We then test AGF-SVGD on Gaussian-Bernoulli restricted Boltzmann machine (RBM) and compare it with advanced gradient-free MCMC such as KAMH (Sejdinovic et al., 2014) and KHMC (Strathmann et al., 2015). Finally, we apply our algorithm to Gaussian process classification on real-world datasets.

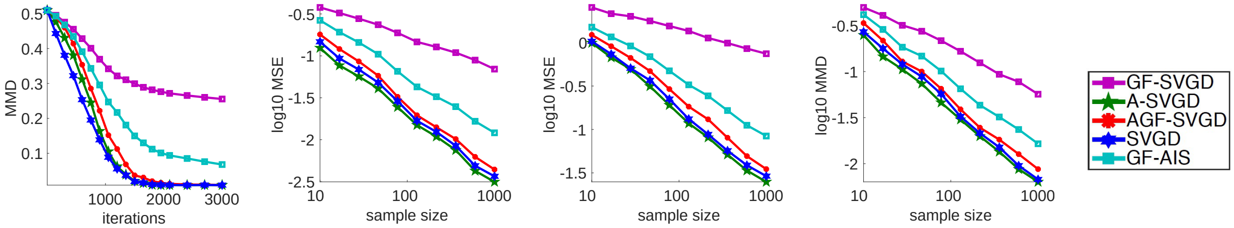
We use RBF kernel $k(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\|\boldsymbol{x} - \boldsymbol{x}'\|^2/h)$ for the updates of our proposed algorithms and the kernel approximation in (24); the bandwidth h is taken to be $h{=}\mathrm{med}^2/(2\log(n+1))$ where med is the median of the current n particles. When maximum mean discrepancy (MMD) (Gretton et al., 2012) is applied to evaluate the sample quality, RBF kernel is used and the bandwidth is chosen based on the median distance of the exact samples so that all methods use the same bandwidth for a fair comparison. Adam optimizer (Kingma & Ba, 2015) is applied to our proposed algorithms for accelerating convergence.

5.1. Simple Gaussian Distributions

We test our basic GF-SVGD in Algorithm 1 on a simple 2D multivariate Gaussian distribution to develop insights on the optimal choice of ρ. We set a Gaussian target $p(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \sigma I)$ with fixed $\boldsymbol{\mu} = (0, 0)$ and $\sigma = 2.0$, and an auxiliary distribution $\rho(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_\rho, \sigma_\rho I)$ where we vary the value of $\boldsymbol{\mu}_\rho$ and σ_ρ in Figure 1. The performance is evaluated based on MMD between GF-SVGD particles and the exact samples from p (Figure 1(a)), and the mean square error (MSE) of estimating $\boldsymbol{\mu}$ and σ (Figure 1(b)-(c)).

Figure 1 suggests a smaller difference in mean $\boldsymbol{\mu}_\rho$ and $\boldsymbol{\mu}$ generally gives better results, but the equal variance $\sigma_\rho = \sigma$ does not achieve the best performance. Instead, it seems that $\sigma_\rho \approx 3\sigma$ gives the best result in this particular case. This suggests by choosing ρ to be a proper distribution that well covers the probability mass of p, it is possible to even outperform the gradient-based SVGD which uses $\rho = p$.

Interestingly, even when we take $\sigma_\rho = \infty$, corresponding to the simple update in (23) with $\rho = 1$, the algorithm still performs reasonably well (although not optimally) in terms of MMD and mean estimation (Figure 1(a)-(b)). It does perform worse on the variance estimation (Figure 1(c)), and we observe that this seems to be because the repulsive force domains when σ_ρ is large (e.g., when $\sigma_\rho = \infty$, only the repulsive term is left as shown in (23)), and it causes the particles to be overly diverse, yielding an over-estimation of the variance. This is interesting because we have found that the standard SVGD with RBF kernel tends to underestimate the variance, and a hybrid of them may be developed to give a more calibrated variance estimation.



(a) Convergence (b) Mean (c) Variance (d) MMD

Figure 2. Results on GMM with 10 random mixture components and 25 dimensions. (a): the convergence of MMD with fixed sample size of $n = 200$. (b)-(c): the MSE vs. sample size when estimating the mean and variance using the particles returned by different algorithms at convergence. (d): the MMD between the particles of different methods and the true distribution p. In (b, c, d), 3000 iterations are used. For GF-AIS, the sample size n represents the number of parallel chains, and the performance is evaluated using the weighted average of the particles at the final iteration with their importance weights given by AIS.

In Figure 1(d), we add additional comparisons with exact Monte Carlo (MC) which directly draws sample from p, and standard importance sampling (IS) with ρ as proposal. We find that GF-SVGD provides much better results than the standard IS strategy with any ρ. In addition, GF-SVGD can even outperform the exact MC and the standard SVGD when auxiliary distribution ρ is chosen properly (e.g., $\sigma_\rho \approx 3\sigma$).

5.2. Gaussian Mixture Models (GMM)

We test GF-SVGD and AGF-SVGD on a 25-dimensional GMM with 10 randomly generated mixture components, $p(\boldsymbol{x}) = \frac{1}{10}\sum_{i=1}^{10}\mathcal{N}(\boldsymbol{x};\boldsymbol{\mu}_i, I)$, with each element of $\boldsymbol{\mu}_i$ is drawn from $\mathrm{Uniform}([-1,1])$. The auxiliary distribution $\rho(\boldsymbol{x})$ is a multivariate Gaussian $\rho(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x};\boldsymbol{\mu}_\rho, \sigma_\rho I)$, with fixed $\sigma_\rho = 4$ and each element of $\boldsymbol{\mu}_\rho$ drawn from $\mathrm{Uniform}([-1,1])$. For AGF-SVGD, we set its initial distribution p_0 to equal the ρ above in GF-SVGD.

Figure 2(a) shows the convergence of MMD vs. the number of iterations of different algorithms with a particle size of $n = 200$, and Figure 2(b)-(d) shows the converged performance as the sample size n varies. It is not surprising to see that that standard SVGD converges fastest since it uses the full gradient information of the target p. A-SVGD converges slightly slower in the beginning but catches up later; this is because that it uses increasingly more gradient information from p. GF-SVGD performs significantly worse, which is expected because it does not leverage the gradient information. However, it is encouraging that annealed GF-SVGD, which also leverages no gradient information, performs much better than GF-SVGD, only slightly worse than the gradient-based SVGD and A-SVGD.

For comparison, we also tested a gradient-free variant of annealed importance sampling (GF-AIS) (Neal, 2001) with a transition probability constructed by Metropolis-adjusted Langevin dynamics, in which we use the same temperature scheme as our AGF-SVGD, and the same surrogate gradient $\nabla_{\boldsymbol{x}} \log \rho_t$ defined in (24). GF-AIS returns a set of particles with importance weights, so we use weighted averages when evaluating the MMD and the mean/variance estimation. This version of GF-AIS is highly comparable to our AGF-SVGD since both of them use the same annealing scheme and surrogate gradient. However, Figure 2 shows that AGF-SVGD still significantly outperforms GF-AIS.

5.3. Gauss-Bernoulli Restricted Boltzmann Machine

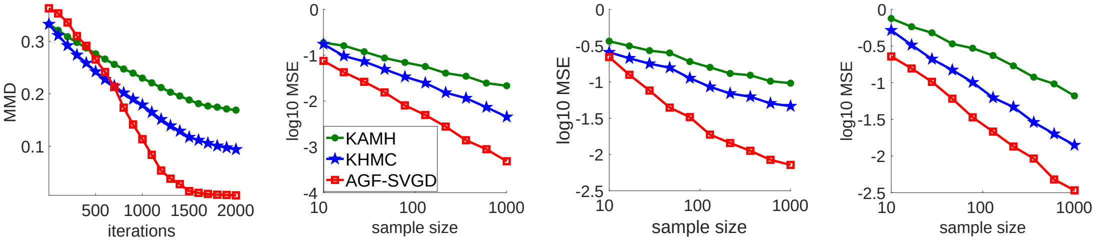
We further compare AGF-SVGD with two recent baselines on Gauss-Bernoulli RBM, defined by

$$p(\boldsymbol{x}) \propto \sum_{\boldsymbol{h}} \exp(\boldsymbol{x}^\top B\boldsymbol{h} + \boldsymbol{c}_1^\top \boldsymbol{x} + \boldsymbol{c}_2^\top \boldsymbol{h} - \frac{1}{2}\|\boldsymbol{x}\|_2^2), \quad (25)$$

where $\boldsymbol{x} \in \mathbb{R}^d$ and $\boldsymbol{h} \in \{\pm 1\}^{d'}$ is a binary latent variable. By marginalizing the hidden variable $\boldsymbol{h}$, we can see that $p(\boldsymbol{x})$ is a special GMM with $2^{d'}$ components. In our experiments, we draw the parameters $\boldsymbol{c}_1$ and $\boldsymbol{c}_2$ from standard Gaussian and select each element of B randomly from $\{\pm 0.5\}$ with equal probabilities. We set the dimension d of $\boldsymbol{x}$ to be 20 and the dimension d' of $\boldsymbol{h}$ to be 10 so that $p(\boldsymbol{x})$ is a 20-dimensional GMM with 2^{10} components, for which it is still feasible to draw exact samples by brute-force for the purpose of evaluation. For AGF-SVGD, we set the initial distribution to be $p_0(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x};\boldsymbol{\mu}, \sigma I)$, with $\boldsymbol{\mu}$ drawn from $\mathrm{Uniform}([1,2])$ and $\sigma = 3$.

We compare our AGF-SVGD with two recent gradient-free methods: KAMH (Sejdinovic et al., 2014) and KHMC (Strathmann et al., 2015). Both methods are advanced MCMC methods that adaptively improves the transition proposals based on kernel-based approximation from the history of Markov chains.

For a fair comparison with SVGD, we run n parallel chains of KAMH and KHMC and take the last samples of n chains for estimation. In addition, we find that both KAMH and KHMC require a relatively long burn-in phase before the adaptive proposal becomes useful. In our experiments, we use 10,000 burn-in steps for both KAMH and KHMC, and



(a) Convergence (b) Mean (c) Variance (d) MMD

Figure 3. Gauss-Bernoulli RBM with $d = 20$ and $d' = 10$. (a): the convergence of MMD with $n = 100$ for all the algorithms. The evaluations of MMD of KAMH and KHMC in (a) starts from the burn-in steps of the typical MH algorithm. (b)-(c): the MSE vs. sample size when estimating the mean and variance using the particles returned by different algorithms at 2000 iterations. (d): the MMD between the particles of different algorithms and the true distribution p at 2000 iterations.

exclude the computation time of burn-in when comparing the convergence speed with GF-SVGD in Figure 3; this gives KAMH and KHMC much advantage for comparison, and the practical computation speed of KAMH and KHMC is much slower than our AGF-SVGD. From Figure 3 (a), we can see that our AGF-SVGD converges fastest to the target p, even when we exclude the 10,000 burn-in steps in KAMH and KHMC. Figure 3 (b, c, d) shows that our AGF-SVGD performs the best in terms of the accuracy of estimating the mean, variance and MMD.

5.4. Gaussian Process Classification

We apply our AGF-SVGD to sample hyper-parameters from marginal posteriors of Gaussian process (GP) binary classification. Consider a classification of predicting binary label $y \in \{\pm 1\}$ from feature $\boldsymbol{z}$. We assume y is generated by a latent Gaussian process $f(\boldsymbol{z})$, $p(y|\boldsymbol{z}) = 1/(1+\exp(-yf(\boldsymbol{z})))$ and f is drawn from a GP prior $f \sim GP(0, k_{f,\boldsymbol{\theta}})$, where $k_{f,\boldsymbol{\theta}}$ is the GP kernel indexed by a hyperparameter $\boldsymbol{\theta}$. In particular, we assume $k_{f,\boldsymbol{\theta}}(\boldsymbol{z}, \boldsymbol{z}') = \exp(-\frac{1}{2}||(\boldsymbol{z}-\boldsymbol{z}')./\exp(\boldsymbol{\theta})||^2)$, where ./ denotes the element-wise division and $\boldsymbol{\theta}$ is a vector of the same size as $\boldsymbol{z}$. Given a dataset $Y = \{y_i\}$ and $Z = \{\boldsymbol{z}_i\}$, we are interested in drawing samples from the posterior distribution $p(\boldsymbol{\theta}|Z, Y)$. Note that the joint posterior of $(\boldsymbol{\theta}, f)$ is

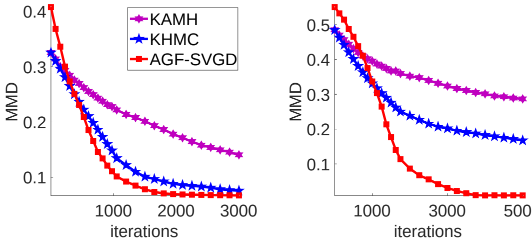
$$p(\boldsymbol{\theta}, f|Z, Y) = p(Y|f, Z)p(f|\boldsymbol{\theta})p(\boldsymbol{\theta}).$$

Since it is intractable to exactly calculate the marginal posterior of $\boldsymbol{\theta}$, we approximate it by

$$\hat{p}(\boldsymbol{\theta}|Z, Y) := p(\boldsymbol{\theta})\frac{1}{m}\sum_{i=1}^{m}\frac{p(Y|f^i, Z)p(f^i|\boldsymbol{\theta})}{q(f^i|\boldsymbol{\theta})}, \qquad (26)$$

where $\{f^i\}_{i=1}^m$ is drawn from a proposal distribution $q(f \mid \boldsymbol{\theta})$, which is constructed by an expectation propagation-based approximation of $p(f|\boldsymbol{\theta}, Z, Y)$ following Filippone & Girolami (2014).

We run multiple standard Metropolis-Hastings chains to obtain ground truth samples from $p(\boldsymbol{\theta} \mid D)$, following the procedures in section 5.1 of Sejdinovic et al. (2014) and Appendix D.3 of Strathmann et al. (2015). We test the algorithms on Glass dataset and SUSY dataset in Figure 4 from UCI repository (Asuncion & Newman, 2007) for which the dimension of $\boldsymbol{\theta}$ is $d = 9$ and $d = 18$, respectively. We initialize our algorithm with draws from $p_0(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \sigma I)$ where $\sigma = 3$ and each element of $\boldsymbol{\mu}$ is drawn from $\mathrm{Uniform}([-1, 1])$. For KAMH and KHMC, we again run n parallel chains and initialize them with an initial burn-in period of 6000 steps which is *not* taken into account in evaluation. Figure 4 shows that AGF-SVGD again converges faster than KAMH and KHMC, even without the additional burn-in period.



(a) Glass dataset (b) SUSY dataset

Figure 4. Sampling from the marginal posteriors on GP classification for Glass dataset (a) and SUSY dataset (b). We use a sample size of $n = 200$ for all methods.

6. Conclusion

We derive a gradient-free extension of Stein's identity and Stein discrepancy and propose a novel gradient-free sampling algorithm. Future direction includes theoretical investigation of optimal choice of the auxiliary proposal with which we may leverage the gradient of the target to further improve the sample efficiency over the standard SVGD. We are also interested in exploring the application of the gradient-free KSD on the goodness-of-fit tests and blackbox importance sampling.

Acknowledgement

This work is supported in part by NSF CRII 1565796.

References

Andrieu, Christophe and Roberts, Gareth O. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, pp. 697–725, 2009.

Asuncion, Arthur and Newman, David. UCI machine learning repository, 2007.

Beaumont, Mark A. Estimation of population growth or decline in genetically monitored populations. *Genetics*, 2003.

Berlinet, Alain and Thomas-Agnan, Christine. *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media, 2011.

Blei, David M, Kucukelbir, Alp, and McAuliffe, Jon D. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 2017.

Cappé, Olivier, Douc, Randal, Guillin, Arnaud, Marin, Jean-Michel, and Robert, Christian P. Adaptive importance sampling in general mixture classes. *Statistics and Computing*, 18(4):447–459, 2008.

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

Chwialkowski, Kacper, Strathmann, Heiko, and Gretton, Arthur. A kernel test of goodness of fit. In *ICML*, 2016.

Cotter, Colin, Cotter, Simon, and Russell, Paul. Parallel adaptive importance sampling. *arXiv:1508.01132*, 2015.

Filippone, Maurizio and Girolami, Mark. Pseudo-marginal Bayesian inference for gaussian processes. *In IEEE*, 2014.

Gilks, Walter R and Wild, Pascal. Adaptive rejection sampling for Gibbs sampling. *Applied Statistics*, 1992.

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

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

Gorham, Jackson and Mackey, Lester. Measuring sample quality with kernels. *arXiv:1703.01717*, 2017.

Gretton, Arthur, Borgwardt, Karsten M, Rasch, Malte J, Schölkopf, Bernhard, and Smola, Alexander. A kernel two-sample test. *JMLR*, 2012.

Han, Jun and Liu, Qiang. Stein variational adaptive importance sampling. *In Uncertainty in Artificial Intelligence*, 2017.

Hoffman, Matthew D and Gelman, Andrew. The no-u-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo. *JMLR*, 15(1):1593–1623, 2014.

Kingma, Diederik and Ba, Jimmy. Adam: A method for stochastic optimization. *ICLR*, 2015.

Liu, Qiang. Stein variational gradient descent as gradient flow. In *NIPS*, 2017.

Liu, Qiang and Lee, Jason D. Black-box importance sampling. *AISTATS*, 2017.

Liu, Qiang and Wang, Dilin. Stein variational gradient descent: A general purpose Bayesian inference algorithm. In *NIPS*, pp. 2370–2378, 2016.

Liu, Qiang, Lee, Jason D, and Jordan, Michael I. A kernelized Stein discrepancy for goodness-of-fit tests. In *ICML*, 2016.

Liu, Yang, Ramachandran, Prajit, Liu, Qiang, and Peng, Jian. Stein variational policy gradient. In *In UAI*, 2017.

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

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

Oates, Chris J, Girolami, Mark, and Chopin, Nicolas. Control functionals for monte carlo integration. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2017.

Sejdinovic, Dino, Strathmann, Heiko, Garcia, Maria Lomeli, Andrieu, Christophe, and Gretton, Arthur. Kernel adaptive Metropolis-Hastings. In *ICML*, 2014.

Strathmann, Heiko, Sejdinovic, Dino, Livingstone, Samuel, Szabo, Zoltan, and Gretton, Arthur. Gradient-free Hamiltonian Monte Carlo with efficient kernel exponential families. In *NIPS*, pp. 955–963, 2015.

Zhang, Cheng, Butepage, Judith, Kjellstrom, Hedvig, and Mandt, Stephan. Advances in variational inference. *arXiv:1711.05597*, 2017.