Sampling with Kernelized Wasserstein Gradient Flows

Anna Korba
ENSAE/CREST

Heilbronn Institute for Mathematical Research

Joint work with Adil Salim (Simons), Giulia Luise (UCL), Michael Arbel (INRIA Grenoble), Arthur Gretton (UCL), Pierre-Cyril Aubin-Frankowski (INRIA Paris), Szymon Majewski (Polytechnique), Pierre Ablin (CNRS).

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

Problem : Sample from a target distribution π over $\mathbb{R}^d$, whose density w.r.t. Lebesgue is known up to a constant Z :

$$\pi(x) = \frac{\tilde{\pi}(x)}{Z}$$

where Z is the (untractable) normalization constant.

Problem : Sample from a target distribution π over $\mathbb{R}^d$, whose density w.r.t. Lebesgue is known up to a constant Z :

$$\pi(x) = \frac{\tilde{\pi}(x)}{Z}$$

where Z is the (untractable) normalization constant.

Motivation : Bayesian statistics.

- Let $\mathcal{D} = (w_i, y_i)_{i=1,\dots,N}$ observed data.
- Assume an underlying model parametrized by θ
  (e.g. $p(y|w,\theta)$ gaussian)

  $\Longrightarrow$ Likelihood: $p(\mathcal{D}|\theta) = \prod_{i=1}^N p(y_i|\theta, w_i)$.
- Assume also $\theta \sim p$ (prior distribution).

Bayes' rule : $\pi(\theta) := p(\theta|\mathcal{D}) = \dfrac{p(\mathcal{D}|\theta)p(\theta)}{Z}$, $Z = \displaystyle\int_{\mathbb{R}^d} p(\mathcal{D}|\theta)p(\theta)d\theta.$

Sampling as optimization over distributions

Assume that $\pi \in \mathcal{P}_2(\mathbb{R}^d) = \{\mu \in \mathcal{P}(\mathbb{R}^d), \int \|x\|^2 d\mu(x) < \infty\}$.
The sampling task can be recast as an optimization problem:

$$\pi = \underset{\mu \in \mathcal{P}_2(\mathbb{R}^d)}{\text{argmin}}\; D(\mu|\pi) := \mathcal{F}(\mu),$$

where D is a **dissimilarity functional**.

Starting from an initial distribution $\mu_0 \in \mathcal{P}_2(\mathbb{R}^d)$, one can then consider the **Wasserstein gradient flow** of $\mathcal{F}$ over $\mathcal{P}_2(\mathbb{R}^d)$ to transport μ_0 to π.

Wasserstein gradient flows [Ambrosio et al., 2008]

The differential of $\mu \mapsto \mathcal{F}(\mu)$ evaluated at $\mu \in \mathcal{P}$ is the unique function $\frac{\partial \mathcal{F}(\mu)}{\partial \mu} : \mathbb{R}^d \to \mathbb{R}$ s. t. for any $\mu, \mu' \in \mathcal{P}$, $\mu' - \mu \in \mathcal{P}$:

$$\lim_{\epsilon \to 0} \frac{1}{\epsilon}(\mathcal{F}(\mu + \epsilon(\mu' - \mu)) - \mathcal{F}(\mu)) = \int_{\mathbb{R}^d} \frac{\partial \mathcal{F}(\mu)}{\partial \mu}(x)(d\mu' - d\mu)(x).$$

Wasserstein gradient flows [Ambrosio et al., 2008]

The differential of $\mu \mapsto \mathcal{F}(\mu)$ evaluated at $\mu \in \mathcal{P}$ is the unique function $\frac{\partial \mathcal{F}(\mu)}{\partial \mu} : \mathbb{R}^d \to \mathbb{R}$ s. t. for any $\mu, \mu' \in \mathcal{P}$, $\mu' - \mu \in \mathcal{P}$:

$$\lim_{\epsilon \to 0} \frac{1}{\epsilon}(\mathcal{F}(\mu + \epsilon(\mu' - \mu)) - \mathcal{F}(\mu)) = \int_{\mathbb{R}^d} \frac{\partial \mathcal{F}(\mu)}{\partial \mu}(x)(d\mu' - d\mu)(x).$$

Then $\mu : [0, \infty] \to \mathcal{P}, t \mapsto \mu_t$ satisfies a Wasserstein gradient flow of $\mathcal{F}$ if distributionnally:

$$\boxed{\frac{\partial \mu_t}{\partial t} = div\left(\mu_t \nabla \frac{\partial \mathcal{F}(\mu_t)}{\partial \mu_t}\right),}$$

where $\nabla_{W_2}\mathcal{F}(\mu) := \nabla \frac{\partial \mathcal{F}(\mu)}{\partial \mu} \in L^2(\mu)$ is called the Wasserstein gradient of $\mathcal{F}$.

Choice of the loss function

Many possibilities for the choice of D among Wasserstein distances, f-divergences, Integral Probability Metrics...

For instance,

- D is the KL (Kullback-Leibler divergence):

$$\mathrm{KL}(\mu|\pi) = \begin{cases} \int_{\mathbb{R}^d} \log\left(\frac{\mu}{\pi}(x)\right) d\mu(x) & \text{if } \mu \ll \pi \\ +\infty & \text{otherwise.} \end{cases}$$

- D is the MMD (Maximum Mean Discrepancy):

$$\mathrm{MMD}^2(\mu,\pi) = \iint_{\mathbb{R}^d} k(x,y) d\mu(x) d\mu(y) + \iint_{\mathbb{R}^d} k(x,y) d\pi(x) d\pi(y) - 2 \iint_{\mathbb{R}^d} k(x,y) d\mu(x) d\pi(y).$$

where $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a p.s.d. kernel.

Two parts for this talk:

- first part : related to the optimization of the KL
- second part : related to the optimization of the MMD

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

Sampling as optimization of the KL

The target distribution π is solution of :

$$\min_{\mu \in \mathcal{P}(\mathbb{R}^d)} \mathrm{KL}(\mu|\pi) \tag{1}$$

Sampling as optimization of the KL

The target distribution π is solution of :

$$\min_{\mu \in \mathcal{P}(\mathbb{R}^d)} \mathrm{KL}(\mu|\pi) \tag{1}$$

1. **Variants of Langevin Monte Carlo (LMC)**

   [Dalalyan, 2017], [Durmus and Moulines, 2016], [Durmus et al., 2019],

   - generates a Markov chain whose law converges to π
   - corresponds to a time-discretization of the gradient flow of the KL
   - rates of convergence deteriorates quickly in high dimensions

Sampling as optimization of the KL

The target distribution π is solution of :

$$\min_{\mu \in \mathcal{P}(\mathbb{R}^d)} \mathrm{KL}(\mu|\pi) \tag{1}$$

1. **Variants of Langevin Monte Carlo (LMC)**

   [Dalalyan, 2017], [Durmus and Moulines, 2016], [Durmus et al., 2019],

   - generates a Markov chain whose law converges to π
   - corresponds to a time-discretization of the gradient flow of the KL
   - rates of convergence deteriorates quickly in high dimensions

2. **Variational Inference (VI):**

   [Alquier and Ridgway, 2017], [Zhang et al., 2018]

   - restrict the search space in (1) to a parametric family
   - tractable in the large scale setting
   - only returns an approximation of π

Sampling as optimization of the KL

The target distribution π is solution of :

$$\min_{\mu \in \mathcal{P}(\mathbb{R}^d)} \mathrm{KL}(\mu|\pi) \tag{1}$$

1. **Variants of Langevin Monte Carlo (LMC)**

   [Dalalyan, 2017], [Durmus and Moulines, 2016], [Durmus et al., 2019],

   - generates a Markov chain whose law converges to π
   - corresponds to a time-discretization of the gradient flow of the KL
   - rates of convergence deteriorates quickly in high dimensions

2. **Variational Inference (VI):**

   [Alquier and Ridgway, 2017], [Zhang et al., 2018]

   - restrict the search space in (1) to a parametric family
   - tractable in the large scale setting
   - only returns an approximation of π

$\Longrightarrow$ Other algorithms can be obtained by discretizing the W_2 gradient flow of the KL...

The KL as a composite functional

$$\mathrm{KL}(\mu|\pi) = \int \log\left(\frac{\mu}{\pi}(x)\right) d\mu(x) \text{ if } \mu \ll \pi, +\infty \text{ else.}$$

The KL as a composite functional

$$\mathrm{KL}(\mu|\pi) = \int \log\left(\frac{\mu}{\pi}(x)\right) d\mu(x) \text{ if } \mu \ll \pi, +\infty \text{ else.}$$

It is written as **a composite functional** :

$$\mathrm{KL}(\mu|\pi) = \underbrace{\int V(x)d\mu(x)}_{\mathcal{E}_V(\mu) \text{ external potential}} + \underbrace{\int \log(\mu(x))d\mu(x)}_{\mathcal{U}(\mu) \text{ negative entropy}} + cte$$

The KL as a composite functional

$$\mathrm{KL}(\mu|\pi) = \int \log\left(\frac{\mu}{\pi}(x)\right) d\mu(x) \text{ if } \mu \ll \pi, +\infty \text{ else.}$$

It is written as **a composite functional** :

$$\mathrm{KL}(\mu|\pi) = \underbrace{\int V(x) d\mu(x)}_{\mathcal{E}_V(\mu) \text{ external potential}} + \underbrace{\int \log(\mu(x)) d\mu(x)}_{\mathcal{U}(\mu) \text{ negative entropy}} + cte$$

W_2 gradient flow of the KL is the Fokker-Planck equation:

$$\frac{\partial \mu_t}{\partial t} = div(\mu_t \underbrace{\nabla \log\left(\frac{\mu_t}{\pi}\right)}_{\nabla_{W_2} \mathrm{KL}(\mu_t|\pi)}) = div(\mu_t \underbrace{\nabla V}_{\nabla_{W_2}\mathcal{E}_V(\mu)}) + div(\mu_t \underbrace{\nabla \log(\mu_t)}_{\mathcal{U}(\mu)})$$

The KL as a composite functional

$$\mathrm{KL}(\mu|\pi) = \int \log\left(\frac{\mu}{\pi}(x)\right) d\mu(x) \text{ if } \mu \ll \pi, +\infty \text{ else.}$$

It is written as **a composite functional** :

$$\mathrm{KL}(\mu|\pi) = \underbrace{\int V(x) d\mu(x)}_{\color{red}{\mathcal{E}_V(\mu) \text{ external potential}}} + \underbrace{\int \log(\mu(x)) d\mu(x)}_{\color{green}{\mathcal{U}(\mu) \text{ negative entropy}}} + cte$$

W_2 gradient flow of the KL is the Fokker-Planck equation:

$$\frac{\partial \mu_t}{\partial t} = div(\mu_t \underbrace{\nabla \log\left(\frac{\mu_t}{\pi}\right)}_{\nabla_{W_2} \mathrm{KL}(\mu_t|\pi)}) = div(\mu_t \underbrace{\nabla V}_{\color{red}{\nabla_{W_2}\mathcal{E}_V(\mu)}}) + div(\mu_t \underbrace{\nabla \log(\mu_t)}_{\color{green}{\mathcal{U}(\mu)}})$$

It is the continuity equation ($X_t \sim \mu_t$) of the Langevin dynamics :

$$dX_t = -\nabla V(X_t) + \sqrt{2} dB_t$$

where (B_t) is the brownian motion in $\mathbb{R}^d$.

Gradient flow of the entropy

The gradient flow of the negative entropy $\mathcal{U}(\mu)$ is the heat equation

$$\frac{\partial \mu_t}{\partial t} = \Delta \mu_t$$

This has an exact solution which is the heat flow $\mu_t = \mu_0 * \mathcal{N}(0, 2tI_d)$.

In space, this is implemented via the addition of Gaussian noise [1]

$$X_t = X_0 + \sqrt{2t}Z \tag{2}$$

where $Z \sim \mathcal{N}(0, I_d)$ and Z independent of X_0.

Some time-discretizations of the KL gradient flow...

[1] The true solution of the heat flow is the Brownian motion in space. However, at each time, the solution has the same distribution as (2)

Unadjusted Langevin Algorithm (ULA)

$$X_{n+1} = X_n - \gamma \nabla V(X_n) + \sqrt{2\gamma}\xi_n \text{ where } \xi_n \sim \mathcal{N}(0, I_d)$$

and $\gamma > 0$ is a **constant** step-size.

Unadjusted Langevin Algorithm (ULA)

$$X_{n+1} = X_n - \gamma \nabla V(X_n) + \sqrt{2\gamma}\xi_n \text{ where } \xi_n \sim \mathcal{N}(0, I_d)$$

and $\gamma > 0$ is a **constant** step-size.

Problem : ULA is biased (has stationary distribution $\pi_\gamma \neq \pi$).

Unadjusted Langevin Algorithm (ULA)

$$X_{n+1} = X_n - \gamma \nabla V(X_n) + \sqrt{2\gamma}\xi_n \quad \text{where } \xi_n \sim \mathcal{N}(0, I_d)$$

and $\gamma > 0$ is a **constant** step-size.

Problem : ULA is biased (has stationary distribution $\pi_\gamma \neq \pi$**).**

We can write ULA as the composition :

$$Y_{n+1} = X_n - \gamma \nabla V(X_n) \quad \text{gradient descent/forward method for V}$$

$$X_{n+1} = Y_{n+1} + \sqrt{2\gamma}\xi_n \quad \text{exact solution for the heat flow}$$

$\Longrightarrow$ Forward-Flow discretization

Unadjusted Langevin Algorithm (ULA)

$$X_{n+1} = X_n - \gamma \nabla V(X_n) + \sqrt{2\gamma}\xi_n \text{ where } \xi_n \sim \mathcal{N}(0, I_d)$$

and $\gamma > 0$ is a **constant** step-size.

Problem : ULA is biased (has stationary distribution $\pi_\gamma \neq \pi$).

We can write ULA as the composition :

$$Y_{n+1} = X_n - \gamma \nabla V(X_n) \quad \text{gradient descent/forward method for V}$$

$$X_{n+1} = Y_{n+1} + \sqrt{2\gamma}\xi_n \quad \text{exact solution for the heat flow}$$

$\Longrightarrow$ Forward-Flow discretization

In the space of measures $\mathcal{P}$:

$$\nu_{n+1} = (I - \gamma \nabla V)_{\#}\mu_n \quad \text{gradient descent for } \mathcal{E}_V$$

$$\mu_{n+1} = \mathcal{N}(0, 2\gamma I) * \nu_{n+1} \quad \text{exact gradient flow for } \mathcal{U}$$

This Forward-flow discretization is biased [Wibisono, 2018]**.**

Other (unbiased) time discretizations

1. Forward method :

$$\mu_{n+1} = exp_{\mu_n}(-\gamma \nabla_{W_2} \mathrm{KL}(\mu_n|\pi)) = \left(I - \gamma \nabla \log\left(\frac{\mu_n}{\pi}\right)\right)_{\#} \mu_n$$

   where $exp_\mu : L^2(\mu) \to \mathcal{P}, \phi \mapsto (I + \phi)_{\#}\mu$,
   and which corresponds in $\mathbb{R}^d$ to:

$$X_{n+1} = X_n - \gamma \nabla \log\left(\frac{\mu_n}{\pi}\right)(X_n) \sim \mu_{n+1}$$

2. Backward method :

$$\mu_{n+1} = JKO_{\gamma \mathrm{KL}(.|\pi)}(\mu_n)$$

$$\text{where } JKO_{\gamma\mathcal{F}}(\nu) = \operatorname*{argmin}_{\mu \in \mathcal{P}} \mathcal{F}(\mu) + \frac{1}{2\gamma} W_2^2(\nu, \mu).$$

3. Forward-Backward method :

$$\nu_{n+1} = (I - \gamma \nabla V)_{\#}\mu_n$$
$$\mu_{n+1} = JKO_{\gamma\mathcal{U}}(\nu_{n+1})$$

Focus on the Forward method

Problem: $\nabla_{W_2} \mathrm{KL}(\mu_n|\pi) = \nabla \log\left(\frac{\mu_n}{\pi}\right)$.

While $\nabla \log \pi$ is known, $\nabla \log \mu_n$ has to be estimated from particles $X_n^1, \dots, X_n^N$, e.g. with[2] :

1. **Kernel Density Estimation (KDE)**:

$$\mu_n(.) \approx \frac{1}{N} \sum_{i=1}^{N} k(X_n^i - .)$$

Then,

$$-\nabla_{W_2} \mathrm{KL}(\mu_n|\pi)(.) \approx -\left(\nabla V(.) + \frac{\sum_{i=1}^N \nabla k(. - X_n^i)}{\sum_{i=1}^N k(. - X_n^i)}\right)$$

Remark : it is not the W_2 gradient of some functional (see the next slide)

[2] assume a symmetric, translation invariant kernel

2. **Blob Method** [Carrillo et al., 2019]:

Instead of

$$\mathcal{U}(\mu) = \int \log(\mu(x)) d\mu(x),$$

consider

$$\mathcal{U}_k(\mu) = \int \log(k \star \mu(x)) d\mu(x), \text{ where } k \star \mu(x) = \int k(x-y) d\mu(y).$$

Then,

$$\frac{\partial \mathcal{U}_k(\mu)}{\partial \mu}(.) = k \star \left(\frac{\mu}{k \star \mu}\right) + \log(k \star \mu)$$

$$\implies \nabla_{W_2} \mathcal{U}_k(\mu) = = \nabla k \star \left(\frac{\mu}{k \star \mu}\right) + \underbrace{\nabla \log(k \star \mu)}_{\frac{\nabla k \star \mu}{k \star \mu}}$$

$$\implies \nabla_{W_2} \mathrm{KL}(\mu_n|\pi)(.) \approx -(\nabla V(.) + \sum_{i=1}^{N} \frac{\nabla k(. - X_n^i)}{\sum_{m=1}^{N} k(X_n^i - X_n^m)} + \frac{\sum_{i=1}^{N} \nabla k(. - X_n^i)}{\sum_{i=1}^{N} k(. - X_n^i)} \Bigg)$$

Stein Variational Gradient Descent

$$-\nabla_{W_2} \mathrm{KL}(\mu_n|\pi)(.) \approx -\frac{1}{N}\left(\sum_{i=1}^{N} k(. - X_n^j)\nabla V(X_n^i) + \nabla_{X_n^i} k(. - X_n^i)\right)$$

3. **Stein Variational Gradient Descent (SVGD)**

[Liu and Wang, 2016],[Liu, 2017], [Duncan et al., 2019]

- "non parametric" VI, only depends on the choice of some kernel k
- corresponds to a time-discretization of the gradient flow of the KL under a metric depending on k
- uses a set of interacting particles to approximate π

`https://chi-feng.github.io/mcmc-demo/app.html?algorithm=HamiltonianMC&target=banana`

SVGD in the ML literature

- **Empirical performance** demonstrated in various tasks:
  - Bayesian inference [Liu and Wang, 2016, Feng et al., 2017, Liu and Zhu, 2018, Detommaso et al., 2018]
  - learning deep probabilistic models [Wang and Liu, 2016, Pu et al., 2017]
  - reinforcement learning [Liu et al., 2017]
- **Theoretical guarantees** :
  - asymptotic theory: (in continuous time, infinite number of particles) converges asymptotically to π [Lu et al., 2019] when V grows at most polynomially
  - non asymptotic theory: no rates of convergence.

This work : non asymptotic analysis of SVGD in the infinite particle regime but discrete time + finite sample approximation.

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

Background on kernels and RKHS [Steinwart and Christmann, 2008]

- Let $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ a positive, semi-definite kernel, e.g.
$$k(x,x') = \exp\left(-\frac{\|x-x'\|^2}{h}\right), \exp\left(-\frac{\|x-x'\|}{h}\right), (c + \|x - x'\|)^\beta \dots$$
- $\mathcal{H}$ its corresponding RKHS (Reproducing Kernel Hilbert Space):
$$\mathcal{H} = \overline{\left\{\sum_{i=1}^m \alpha_i k(\cdot, x_i);\ m \in \mathbb{N};\ \alpha_1, \dots, \alpha_m \in \mathbb{R};\ x_1, \dots, x_m \in \mathbb{R}^d\right\}}$$
- $\mathcal{H}$ is a Hilbert space with inner product $\langle ., . \rangle_{\mathcal{H}}$ and norm $\|.\|_{\mathcal{H}}$.
  It satisfies the reproducing property:
$$\forall \quad f \in \mathcal{H},\ x \in \mathbb{R}^d, \quad f(x) = \langle f, k(x,.)\rangle_{\mathcal{H}}$$

We assume $\int_{\mathbb{R}^d \times \mathbb{R}^d} k(x,x) d\mu(x) < \infty$ for any $\mu \in \mathcal{P}$. $\implies \mathcal{H} \subset L^2(\mu)$.

For instance assume $\|k(x,.)\|^2_{\mathcal{H}_k} = k(x,x) \leq B^2$, then for $f \in \mathcal{H}_k$
$$\|f\|^2_{L^2(\mu)} = \int \|f(x)\|^2 d\mu(x) = \int \langle f, k(x,.)\rangle^2_{\mathcal{H}_k} d\mu(x)$$
$$\leq \|f\|^2_{\mathcal{H}_k} \int k(x,x) d\mu(x) \leq B^2 \|f\|^2_{\mathcal{H}_k}$$

The kernel integral operator

Then, the inclusion from $\iota : \mathcal{H} \to L^2(\mu)$ admits an adjoint $\iota^\star = S_\mu$, where $S_\mu : L^2(\mu) \to \mathcal{H}$ is defined by:

$$\boxed{S_\mu f(\cdot) = \int k(x, .) f(x) d\mu(x), \quad f \in L^2(\mu).}$$

The kernel integral operator

Then, the inclusion from $\iota : \mathcal{H} \to L^2(\mu)$ admits an adjoint $\iota^\star = S_\mu$, where $S_\mu : L^2(\mu) \to \mathcal{H}$ is defined by:

$$\boxed{S_\mu f(\cdot) = \int k(x,.)f(x)d\mu(x), \quad f \in L^2(\mu).}$$

We have for any $f, g \in L_2(\mu) \times \mathcal{H}$:

$$\langle f, \iota g \rangle_{L^2(\mu)} = \langle \iota^* f, g \rangle_{\mathcal{H}} = \langle S_\mu f, g \rangle_{\mathcal{H}}.$$

We will denote $P_\mu = \iota \circ S_\mu$.

SVGD algorithm

SVGD trick: applying this operator to the W_2 gradient of $\mathrm{KL}(\cdot|\pi)$ leads to

$$P_\mu \nabla \log\left(\frac{\mu}{\pi}\right)(\cdot) = -\int [\nabla \log \pi(x) k(x,\cdot) + \nabla_x k(x,\cdot)] d\mu(x),$$

under appropriate boundary conditions on k and π, e.g. $\lim_{\|x\|\to\infty} k(x,\cdot)\pi(x) \to 0$.

SVGD algorithm

SVGD trick: applying this operator to the W_2 gradient of $\mathrm{KL}(\cdot|\pi)$ leads to

$$P_\mu \nabla \log\left(\frac{\mu}{\pi}\right)(\cdot) = -\int [\nabla \log \pi(x) k(x,\cdot) + \nabla_x k(x,\cdot)] d\mu(x),$$

under appropriate boundary conditions on k and π, e.g. $\lim_{\|x\|\to\infty} k(x,\cdot)\pi(x) \to 0$.

Algorithm : Starting from N i.i.d. samples $(X_0^i)_{i=1,\dots,N} \sim \mu_0$, SVGD algorithm updates the N particles as follows :

$$X_{n+1}^i = X_n^i - \gamma \underbrace{\left[\frac{1}{N}\sum_{j=1}^{N} k(X_n^i, X_n^j)\nabla_{X_n^j} \log \pi(X_n^j) + \nabla_{X_n^j} k(X_n^j, X_n^i)\right]}_{P_{\hat{\mu}_n}\nabla \log\left(\frac{\hat{\mu}_n}{\pi}\right)(X_n^i), \quad \text{with } \hat{\mu}_n = \frac{1}{N}\sum_{j=1}^{N} \delta_{X_n^j}}$$

SVGD algorithm

SVGD trick: applying this operator to the W_2 gradient of $\mathrm{KL}(\cdot|\pi)$ leads to

$$P_\mu \nabla \log\left(\frac{\mu}{\pi}\right)(\cdot) = -\int [\nabla \log \pi(x) k(x,\cdot) + \nabla_x k(x,\cdot)] d\mu(x),$$

under appropriate boundary conditions on k and π, e.g. $\lim_{\|x\|\to\infty} k(x,\cdot)\pi(x) \to 0$.

Algorithm : Starting from N i.i.d. samples $(X_0^i)_{i=1,\dots,N} \sim \mu_0$, SVGD algorithm updates the N particles as follows :

$$X_{n+1}^i = X_n^i - \gamma \underbrace{\left[\frac{1}{N}\sum_{j=1}^N k(X_n^i, X_n^j)\nabla_{X_n^j} \log \pi(X_n^j) + \nabla_{X_n^j} k(X_n^j, X_n^i)\right]}_{P_{\hat{\mu}_n}\nabla \log\left(\frac{\hat{\mu}_n}{\pi}\right)(X_n^i), \quad \text{with } \hat{\mu}_n = \frac{1}{N}\sum_{j=1}^N \delta_{X_n^j}}$$

This work : non asymptotic analysis of SVGD in the infinite particle regime + finite sample approximation.

Continuous-time dynamics of SVGD

SVGD gradient flow [Liu, 2017],[Lu et al., 2019]:

$$\frac{\partial \mu_t}{\partial t} + div(\mu_t V_t) = 0, \qquad V_t := -P_{\mu_t} \nabla \log \left(\frac{\mu_t}{\pi} \right)$$

Continuous-time dynamics of SVGD

SVGD gradient flow [Liu, 2017],[Lu et al., 2019]:

$$\frac{\partial \mu_t}{\partial t} + div(\mu_t V_t) = 0, \qquad V_t := -P_{\mu_t} \nabla \log\left(\frac{\mu_t}{\pi}\right)$$

How fast the KL decreases along SVGD dynamics?

$$\begin{aligned}
\frac{d\,\mathrm{KL}(\mu_t|\pi)}{dt} &= \left\langle V_t, \nabla \log\left(\frac{\mu_t}{\pi}\right)\right\rangle_{L^2(\mu_t)} \\
&= -\left\langle \iota S_{\mu_t} \nabla \log\left(\frac{\mu_t}{\pi}\right), \nabla \log\left(\frac{\mu_t}{\pi}\right)\right\rangle_{L^2(\mu_t)} \\
&= -\underbrace{\left\| S_{\mu_t} \nabla \log\left(\frac{\mu_t}{\pi}\right)\right\|^2_{\mathcal{H}}}_{\mathrm{KSD}^2(\mu_t|\pi)} \text{ since } \iota^* = S_{\mu_t} \\
&\leq 0.
\end{aligned}$$

Continuous-time dynamics of SVGD

SVGD gradient flow [Liu, 2017],[Lu et al., 2019]:

$$\frac{\partial \mu_t}{\partial t} + div(\mu_t V_t) = 0, \qquad V_t := -P_{\mu_t} \nabla \log\left(\frac{\mu_t}{\pi}\right)$$

How fast the KL decreases along SVGD dynamics?

$$\begin{aligned}
\frac{d\,\mathrm{KL}(\mu_t|\pi)}{dt} &= \left\langle V_t, \nabla \log\left(\frac{\mu_t}{\pi}\right)\right\rangle_{L^2(\mu_t)} \\
&= -\left\langle \iota S_{\mu_t} \nabla \log\left(\frac{\mu_t}{\pi}\right), \nabla \log\left(\frac{\mu_t}{\pi}\right)\right\rangle_{L^2(\mu_t)} \\
&= -\underbrace{\left\| S_{\mu_t} \nabla \log\left(\frac{\mu_t}{\pi}\right)\right\|^2_{\mathcal{H}}}_{\mathrm{KSD}^2(\mu_t|\pi)} \text{ since } \iota^* = S_{\mu_t} \\
&\leq 0.
\end{aligned}$$

On the r.h.s. we have the squared **Kernel Stein discrepancy (KSD)** [Chwialkowski et al., 2016] or **Stein Fisher information** at μ_t.

Stein Fisher information

Stationary condition : $\mathrm{KSD}^2(\mu_t|\pi) = \left\| S_{\mu_t} \nabla \log\left(\frac{\mu_t}{\pi}\right)\right\|^2_{\mathcal{H}} = 0.$

Implies weak convergence of μ_t to π if [Gorham and Mackey, 2017]:

- π is distantly dissipative[3] (e.g. gaussian mixtures)
- k is translation invariant with a non-vanishing Fourier transform;
  or k is the IMQ kernel defined by $k(x,y) = (c^2 + \|x-y\|_2^2)^\beta$ for $c > 0$ and $\beta \in [-1, 0]$ (slow decay rate).

[3] $\liminf_{r\to\infty} \kappa(r) > 0$ for $\kappa(r) = \inf\{-2\langle \nabla \log \pi(x) - \nabla \log \pi(y), x-y\rangle / \|x-y\|_2^2 ; \|x-y\|_2^2 = r\}$

Stein Fisher information

Stationary condition : $\mathrm{KSD}^2(\mu_t|\pi) = \left\| S_{\mu_t} \nabla \log\left(\frac{\mu_t}{\pi}\right) \right\|^2_{\mathcal{H}} = 0.$

Implies weak convergence of μ_t to π if [Gorham and Mackey, 2017]:

- π is distantly dissipative[3] (e.g. gaussian mixtures)
- k is translation invariant with a non-vanishing Fourier transform;
  or k is the IMQ kernel defined by $k(x,y) = (c^2 + \|x-y\|_2^2)^\beta$ for $c > 0$ and $\beta \in [-1, 0]$ (slow decay rate).

We show that if k is bounded, $\pi \propto \exp(-V)$ with H_V bounded above and if $\exists C > 0$, $\int \|x\|^2 d\mu_t(x) < C$ for all $t > 0$, then $\mathrm{KSD}^2(\mu_t|\pi) \to 0$.

[3] $\liminf_{r\to\infty} \kappa(r) > 0$ for $\kappa(r) = \inf\{-2\langle \nabla \log \pi(x) - \nabla \log \pi(y), x-y\rangle / \|x-y\|_2^2 ; \|x-y\|_2^2 = r\}$

Convergence of continuous-time dynamics

The convergence of the Stein Fisher information to 0 can be slow. **When do we have fast convergence of SVGD dynamics?**

π satisfies the Stein log-Sobolev inequality [Duncan et al., 2019] with constant $\lambda > 0$ if for any μ:

$$\mathrm{KL}(\mu|\pi) \leq \frac{1}{2\lambda} \mathrm{KSD}^2(\mu|\pi).$$

Convergence of continuous-time dynamics

The convergence of the Stein Fisher information to 0 can be slow. **When do we have fast convergence of SVGD dynamics?**

π satisfies the Stein log-Sobolev inequality [Duncan et al., 2019] with constant $\lambda > 0$ if for any μ:

$$\mathrm{KL}(\mu|\pi) \leq \frac{1}{2\lambda} \mathrm{KSD}^2(\mu|\pi).$$

If it holds,

$$\frac{d\,\mathrm{KL}(\mu_t|\pi)}{dt} = -\,\mathrm{KSD}^2(\mu_t|\pi) \leq -2\lambda\,\mathrm{KL}(\mu_t|\pi)$$

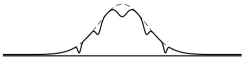
and by integrating :

$$\mathrm{KL}(\mu_t|\pi) \leq e^{-2\lambda t}\,\mathrm{KL}(\mu_0|\pi).$$

"Classic" log-Sobolev inequality upper bounds the KL by the Fisher divergence :

$$\mathrm{KL}(\mu|\pi) \leq \frac{1}{2\lambda} \left\| \nabla \log \left(\frac{\mu}{\pi} \right) \right\|_{L^2(\mu)}^2 .$$

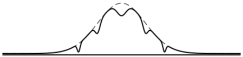
satisfied as soon as π is λ-log concave, but it's more general.



"Classic" log-Sobolev inequality upper bounds the KL by the Fisher divergence :

$$\mathrm{KL}(\mu|\pi) \leq \frac{1}{2\lambda} \left\| \nabla \log\left(\frac{\mu}{\pi}\right) \right\|^2_{L^2(\mu)}.$$

satisfied as soon as π is λ-log concave, but it's more general.



When is Stein log-Sobolev satisfied? not as well known and understood [Duncan et al., 2019], but :

- it fails to hold if k is too regular with respect to π
- some working examples in dimension 1
- whether it holds in higher dimension is more challenging and subject to further research...

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

Proof of a descent lemma for GD of a smooth function

Gradient descent for $V : \mathbb{R}^d \to \mathbb{R}$ a $C^2(\mathbb{R}^d)$ s.t. $\|H_V(x)\| \leq M$ for any x.

$$x_{n+1} = x_n - \gamma \nabla V(x_n).$$

Proof of a descent lemma for GD of a smooth function

Gradient descent for $V : \mathbb{R}^d \to \mathbb{R}$ a $C^2(\mathbb{R}^d)$ s.t. $\|H_V(x)\| \leq M$ for any x.

$$x_{n+1} = x_n - \gamma \nabla V(x_n).$$

Denote $x(t) = x_n - t\nabla V(x_n)$ and $\varphi(t) = V(x(t))$. Using Taylor expansion :

$$\varphi(\gamma) = \varphi(0) + \gamma\varphi'(0) + \int_0^\gamma (\gamma - t)\varphi''(t)dt.$$

Proof of a descent lemma for GD of a smooth function

Gradient descent for $V : \mathbb{R}^d \to \mathbb{R}$ a $C^2(\mathbb{R}^d)$ s.t. $\|H_V(x)\| \leq M$ for any x.

$$x_{n+1} = x_n - \gamma \nabla V(x_n).$$

Denote $x(t) = x_n - t\nabla V(x_n)$ and $\varphi(t) = V(x(t))$. Using Taylor expansion :

$$\varphi(\gamma) = \varphi(0) + \gamma\varphi'(0) + \int_0^\gamma (\gamma - t)\varphi''(t)dt.$$

Since ($\ddot{x}(t) = 0$):

$$\varphi'(0) = \langle \nabla V(x(0)), \dot{x}(0)\rangle = \langle \nabla V(x(0)), -\nabla V(x_n)\rangle = -\|\nabla V(x_n)\|^2,$$

$$\varphi''(t) = \langle \dot{x}(t), H_V(x(t))\dot{x}(t)\rangle \leq M\|\dot{x}(t)\|^2 = M\|\nabla V(x_n)\|^2,$$

Proof of a descent lemma for GD of a smooth function

Gradient descent for $V : \mathbb{R}^d \to \mathbb{R}$ a $C^2(\mathbb{R}^d)$ s.t. $\|H_V(x)\| \leq M$ for any x.

$$x_{n+1} = x_n - \gamma \nabla V(x_n).$$

Denote $x(t) = x_n - t\nabla V(x_n)$ and $\varphi(t) = V(x(t))$. Using Taylor expansion :

$$\varphi(\gamma) = \varphi(0) + \gamma\varphi'(0) + \int_0^\gamma (\gamma - t)\varphi''(t)dt.$$

Since ($\ddot{x}(t) = 0$):

$$\varphi'(0) = \langle \nabla V(x(0)), \dot{x}(0)\rangle = \langle \nabla V(x(0)), -\nabla V(x_n)\rangle = -\|\nabla V(x_n)\|^2,$$

$$\varphi''(t) = \langle \dot{x}(t), H_V(x(t))\dot{x}(t)\rangle \leq M\|\dot{x}(t)\|^2 = M\|\nabla V(x_n)\|^2,$$

we have

$$V(x_{n+1}) \leq V(x_n) - \gamma\|\nabla V(x_n)\|^2 + M\int_0^\gamma (\gamma - t)\|\nabla V(x_n)\|^2 dt$$

$$V(x_{n+1}) - V(x_n) \leq -\gamma\left(1 - \frac{M\gamma}{2}\right)\|\nabla V(x_n)\|^2.$$

A descent lemma for SVGD

Recall that $\pi \propto \exp(-V)$ and assume $\|H_V(x)\| \leq M$. Here, the Hessian of the KL at μ is an operator on $L^2(\mu)$ where:

$$\langle f, Hess_{\mathrm{KL}(.|\pi)}(\mu) f\rangle_{L^2(\mu)} = \mathbb{E}_{X\sim\mu}\left[\langle f(X), H_V(X) f(X)\rangle + \|Jf(X)\|_{HS}^2\right]$$

and yet, this operator **is not bounded** due to the Jacobian term.

A descent lemma for SVGD

Recall that $\pi \propto \exp(-V)$ and assume $\|H_V(x)\| \leq M$. Here, the Hessian of the KL at μ is an operator on $L^2(\mu)$ where:

$$\langle f, Hess_{\mathrm{KL}(.|\pi)}(\mu) f \rangle_{L^2(\mu)} = \mathbb{E}_{X\sim\mu}\left[\langle f(X), H_V(X) f(X)\rangle + \|Jf(X)\|^2_{HS}\right]$$

and yet, this operator **is not bounded** due to the Jacobian term.

In the case of SVGD, one restricts the descent directions f to $\mathcal{H}$. Under several assumptions (boundedness of k and ∇k, of Hessian of V and moments on the trajectory) we could show for γ small enough:

$$\mathrm{KL}(\mu_{n+1}|\pi) - \mathrm{KL}(\mu_n|\pi) \leq -c_\gamma \underbrace{\left\| S_{\mu_n} \nabla \log\left(\frac{\mu_n}{\pi}\right)\right\|^2_{\mathcal{H}}}_{\mathrm{KSD}^2(\mu_n|\pi)}.$$

Sketch of proof - 1

Fix $n \geq 0$. Denote $g = P_{\mu_n} \nabla \log\left(\frac{\mu_n}{\pi}\right)$, $\phi_t = I - tg$ for $t \in [0, \gamma]$ and $\rho_t = (\phi_t)_{\#}\mu_n$. We have $\frac{\partial \rho_t}{\partial t} = div(\rho_t w_t)$ with $w_t = -g \circ \phi_t^{-1}$.

Denote $\varphi(t) = \mathrm{KL}(\rho_t|\pi)$. Using a Taylor expansion,

$$\varphi(\gamma) = \varphi(0) + \gamma\varphi'(0) + \int_0^\gamma (\gamma - t)\varphi''(t)dt.$$

Step 1. $\varphi(0) = \mathrm{KL}(\mu_n|\pi)$ and $\varphi(\gamma) = \mathrm{KL}(\mu_{n+1}|\pi)$.

Step 2. Using the chain rule,

$$\varphi'(t) = \langle \nabla_{W_2} \mathrm{KL}(\rho_t|\pi), w_t \rangle_{L^2(\rho_t)}.$$

Hence :

$$\varphi'(0) = -\langle \nabla \log\left(\frac{\mu_n}{\pi}\right), g \rangle_{L^2(\mu_n)} = -\left\| S_{\mu_n} \nabla \log\left(\frac{\mu_n}{\pi}\right) \right\|_{\mathcal{H}}^2 .$$

Sketch of proof - 2

Step 3.

$$\varphi''(t) = \langle w_t, Hess_{\mathrm{KL}(.|\pi)}(\rho_t) w_t \rangle_{L^2(\rho_t)} := \psi_1(t) + \psi_2(t),$$

$$\psi_1(t) = \mathbb{E}_{x\sim\rho_t}\left[\langle w_t(x), H_V(x) w_t(x)\rangle\right] \text{ and } \psi_2(t) = \mathbb{E}_{x\sim\rho_t}\left[\| J w_t(x) \|^2_{HS}\right]$$

where $\rho_t = (\phi_t)_{\#}\mu_n$, $w_t = -g \circ (\phi_t)^{-1}$.

Sketch of proof - 2

Step 3.

$$\varphi''(t) = \langle w_t, Hess_{\mathrm{KL}(.|\pi)}(\rho_t) w_t \rangle_{L^2(\rho_t)} := \psi_1(t) + \psi_2(t),$$

$$\psi_1(t) = \mathbb{E}_{x\sim\rho_t}\left[\langle w_t(x), H_V(x) w_t(x)\rangle\right] \text{ and } \psi_2(t) = \mathbb{E}_{x\sim\rho_t}\left[\|Jw_t(x)\|_{HS}^2\right]$$

where $\rho_t = (\phi_t)_{\#}\mu_n$, $w_t = -g \circ (\phi_t)^{-1}$.

Step 3.a. Assuming $\|H_V\| \leq M$ and $k(.,.) \leq B$:

$$\psi_1(t) \leq M\|g\|_{L^2(\mu_n)}^2 \leq MB^2 \left\| S_{\mu_n} \nabla \log\left(\frac{\mu_n}{\pi}\right) \right\|_{\mathcal{H}}^2.$$

Sketch of proof - 2

Step 3.

$$\varphi''(t) = \langle w_t, Hess_{\mathrm{KL}(.|\pi)}(\rho_t) w_t \rangle_{L^2(\rho_t)} := \psi_1(t) + \psi_2(t),$$

$$\psi_1(t) = \mathbb{E}_{x\sim\rho_t}\left[\langle w_t(x), H_V(x) w_t(x)\rangle\right] \text{ and } \psi_2(t) = \mathbb{E}_{x\sim\rho_t}\left[\|Jw_t(x)\|_{HS}^2\right]$$

where $\rho_t = (\phi_t)_{\#}\mu_n$, $w_t = -g \circ (\phi_t)^{-1}$.

Step 3.a. Assuming $\|H_V\| \leq M$ and $k(.,.) \leq B$:

$$\psi_1(t) \leq M\|g\|_{L^2(\mu_n)}^2 \leq MB^2 \left\|S_{\mu_n}\nabla \log\left(\frac{\mu_n}{\pi}\right)\right\|_{\mathcal{H}}^2.$$

Step 3.b. Since $\rho_t = (\phi_t)_{\#}\mu_n$, $w_t = -g \circ (\phi_t)^{-1}$,

$$\begin{aligned}\psi_2(t) = \mathbb{E}_{x\sim\mu_n}[\|Jw_t \circ \phi_t(x)\|_{HS}^2] &\leq \|Jg(x)\|_{HS}^2 \|(J\phi_t)^{-1}(x)\|_{op}^2 \\ &\leq B^2 \left\|S_{\mu_n}\nabla \log\left(\frac{\mu_n}{\pi}\right)\right\|_{\mathcal{H}}^2 \alpha^2,\end{aligned}$$

assuming $\|\nabla k(.,.)\| \leq B$ and choosing $\gamma \leq f(\alpha)$ with $\alpha > 1$.

From:

$$\varphi(\gamma) = \varphi(0) + \gamma\varphi'(0) + \int_0^\gamma (\gamma - t)\varphi''(t)dt$$

we have:

$$\begin{aligned}\mathrm{KL}(\mu_{n+1}|\pi) - \mathrm{KL}(\mu_n|\pi) \leq & -\gamma\|S_{\mu_n}\nabla \log\left(\frac{\mu_n}{\pi}\right)\|^2_{\mathcal{H}} \\ & + \frac{\gamma^2}{2}(\alpha^2 + M)B^2\|S_{\mu_n}\nabla \log\left(\frac{\mu_n}{\pi}\right)\|^2_{\mathcal{H}}.\end{aligned}$$

Choosing γ small enough yields a descent lemma :

$$\mathrm{KL}(\mu_{n+1}|\pi) - \mathrm{KL}(\mu_n|\pi) \leq -c_\gamma \underbrace{\left\|S_{\mu_n}\nabla \log\left(\frac{\mu_n}{\pi}\right)\right\|^2_{\mathcal{H}}}_{\mathrm{KSD}^2(\mu_n|\pi)}.$$

Rates in terms of the Stein Fisher Information

Consequence of the descent lemma: for γ small enough,

$$\boxed{\min_{k=1,\dots,n} \mathrm{KSD}^2(\mu_n|\pi) \leq \frac{1}{n}\sum_{k=1}^{n} \mathrm{KSD}^2(\mu_k|\pi) \leq \frac{\mathrm{KL}(\mu_0|\pi)}{c_\gamma n}.}$$

This result does not rely on:

- convexity of V
- nor on Stein log Sobolev inequality
- but only on smoothness of V.

unlike most convergence results on LMC which rely on Log Sobolev inequality or convexity of V.

Rates in terms of the KL objective?

To obtain rates, one may combine a descent lemma (1) of the form

$$\mathrm{KL}(\mu_{n+1}|\pi) - \mathrm{KL}(\mu_n|\pi) \leq -c_\gamma \left\| S_{\mu_n} \nabla \log\left(\frac{\mu_n}{\pi}\right) \right\|_{\mathcal{H}}^2$$

and the Stein log-Sobolev inequality (2) with constant λ:

$$\mathrm{KL}(\mu_{n+1}|\pi) - \mathrm{KL}(\mu_n|\pi) \underbrace{\leq}_{(1)} -c_\gamma \left\| S_{\mu_n} \nabla \log\left(\frac{\mu_n}{\pi}\right) \right\|_{\mathcal{H}}^2 \underbrace{\leq}_{(2)} -c_\gamma 2\lambda KL(\mu_n|\pi).$$

Iterating this inequality yields $\mathrm{KL}(\mu_n|\pi) \leq (1 - 2c_\gamma\lambda)^n \,\mathrm{KL}(\mu_0|\pi)$.

"Classic" approach in optimization [Karimi et al., 2016] or in the analysis of LMC.

Not possible to combine both....

Given that both the kernel and its derivative are bounded, the equation

$$\int \sum_{i=1}^{d} [(\partial_i V(x))^2 k(x,x) \\ - \partial_i V(x)(\partial_i^1 k(x,x) + \partial_i^2 k(x,x)) + \partial_i^1 \partial_i^2 k(x,x)] d\pi(x) < \infty \quad (3)$$

reduces to a property on V which, as far as we can tell, always holds on $\mathbb{R}^d$...

Not possible to combine both....

Given that both the kernel and its derivative are bounded, the equation

$$\int \sum_{i=1}^{d} [(\partial_i V(x))^2 k(x,x) \\ - \partial_i V(x)(\partial_i^1 k(x,x) + \partial_i^2 k(x,x)) + \partial_i^1 \partial_i^2 k(x,x)] d\pi(x) < \infty \quad (3)$$

reduces to a property on V which, as far as we can tell, always holds on $\mathbb{R}^d$...

and this implies that Stein LSI does not hold [Duncan et al., 2019].

Remark : Equation (3) does not hold for :

- k polynomial of order ≥ 3, and
- π with exploding β moments with $\beta \geq 3$ (ex: a student distribution, which belongs to $\mathcal{P}$ the set of distributions with bounded second moment).

Experiments

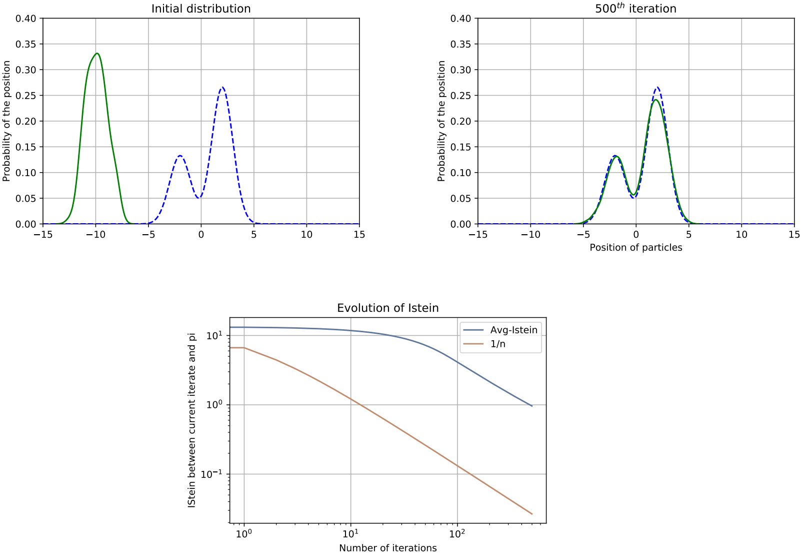


Figure: The particle implementation of the SVGD algorithm illustrates the convergence of $\mathrm{KSD}^2(\mu_n|\pi)$ to 0.

We already have a bound on μ_n versus π. What about $\hat{\mu}_n$?

Recall that the practical SVGD implementation is :

$$X_{n+1}^i = X_n^i - \gamma P_{\hat{\mu}_n} \nabla \log \left(\frac{\hat{\mu}_n}{\pi} \right) (X_n^i), \qquad \hat{\mu}_n = \frac{1}{N} \sum_{j=1}^{N} \delta_{X_n^j}.$$

where $\hat{\mu}_n$ denotes the empirical distribution of the interacting particles.

We already have a bound on μ_n versus π. What about $\hat{\mu}_n$?

Recall that the practical SVGD implementation is :

$$X_{n+1}^i = X_n^i - \gamma P_{\hat{\mu}_n} \nabla \log\left(\frac{\hat{\mu}_n}{\pi}\right)(X_n^i), \qquad \hat{\mu}_n = \frac{1}{N}\sum_{j=1}^N \delta_{X_n^j}.$$

where $\hat{\mu}_n$ denotes the empirical distribution of the interacting particles.

Propagation of chaos result
Let $n \geq 0$ and $T > 0$. Under boundedness and Lipschitzness assumptions for all $k, \nabla k, V$; for any $0 \leq n \leq \frac{T}{\gamma}$ we have :

$$\mathbb{E}[W_2^2(\bar{\mu}_n, \hat{\mu}_n)] \leq \frac{1}{2}\left(\frac{1}{\sqrt{N}}\sqrt{var(\mu_0)}e^{LT}\right)(e^{2LT} - 1)$$

where L is a constant depending on k and π and $\bar{\mu}_n = \frac{1}{N}\sum_{j=1}^N \delta_{\bar{X}_n^j}$ with $\bar{X}_n^j \sim \mu_n$ i.i.d.

Contributions and openings

- First rates of convergence for SVGD, using techniques from optimal transport and optimization (discrete time - infinite number of particles)
- Propagation of chaos bound (finite number of particles regime)

Open questions

- Rates in KL?
- Propagation of chaos : weaker assumptions? uniform in time (UIT)?
- Is it possible to obtain a unified convergence bound (decreasing as $n, N \to \infty$)? (requires UIT)

$$D(\widehat{\mu}_n, \pi) \leq A_n + B_N$$

- how good is SVGD quantisation?
- Other kernels?
  SVGD dynamics also appear in black-box variational inference and Gans [Chu et al., 2020], where the kernel is *the neural tangent kernel* and **depends on the current distribution** ($k \Longrightarrow k_{\mu_n}$)

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

Recall that

$$\pi = \underset{\mu \in \mathcal{P}_2(\mathbb{R}^d)}{\operatorname{argmin}} \; D(\mu|\pi) := \mathcal{F}(\mu),$$

where D is a **dissimilarity functional.**
Here we choose D as the **Kernel Stein Discrepancy (KSD).**

We propose an algorithm that is:

- score-based (only requires $\nabla \log \pi$)
- using a set of particles whose empirical distribution minimizes the KSD
- easy to implement and to use (e.g. leverages L-BFGS) !

We study:

- its convergence properties (numerically and theoretically)
- its empirical performance compared to Stein Variational Gradient Descent

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

Kernel Stein Discrepancy [Chwialkowski et al., 2016, Liu et al., 2016]

For $\mu, \pi \in \mathcal{P}_2(\mathbb{R}^d)$, the KSD of μ relative to π is defined as

$$\mathrm{KSD}^2(\mu|\pi) = \iint k_\pi(x,y)d\mu(x)d\mu(y),$$

where $k_\pi : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is the **Stein kernel**, defined through

- the score function $s(x) = \nabla \log \pi(x)$,
- a p.s.d. kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, $k \in C^2(\mathbb{R}^d)$[4]

For $x, y \in \mathbb{R}^d$,

$$\begin{aligned} k_\pi(x,y) &= s(x)^T s(y) k(x,y) + s(x)^T \nabla_2 k(x,y) \\ &\quad + \nabla_1 k(x,y)^T s(y) + \boldsymbol{\nabla} \cdot_1 \nabla_2 k(x,y) \\ &= \sum_{i=1}^d \frac{\partial \log \pi(x)}{\partial x_i} . \frac{\partial \log \pi(y)}{\partial y_i} . k(x,y) + \frac{\partial \log \pi(x)}{\partial x_i} . \frac{\partial k(x,y)}{\partial y_i} \\ &\quad + \frac{\partial \log \pi(y)}{\partial y_i} . \frac{\partial k(x,y)}{\partial x_i} + \frac{\partial^2 k(x,y)}{\partial x_i \partial y_i} \in \mathbb{R}. \end{aligned}$$

[4] e.g. : $k(x,y) = \exp\left(-\|x-y\|^2/h\right)$

We have seen that the KSD^2 is also as a kernelized Fisher divergence ($\left\|\nabla \log\left(\frac{\mu}{\pi}\right)\right\|^2_{L^2(\mu)}$):

$$\mathrm{KSD}^2(\mu|\pi) = \left\|S_{\mu,k}\nabla \log\left(\frac{\mu}{\pi}\right)\right\|^2_{\mathcal{H}_k}, \; S_{\mu,k} : f \mapsto \int f(x)k(x,.)d\mu(x).$$

$$\begin{aligned}\left\|S_{\mu,k}\nabla \log\left(\frac{\mu}{\pi}\right)\right\|^2_{\mathcal{H}_k} &= \langle S_{\mu,k}\nabla \log\left(\frac{\mu}{\pi}\right), S_{\mu,k}\nabla \log\left(\frac{\mu}{\pi}\right)\rangle_{\mathcal{H}_k} \\ &= \int\int \nabla \log\left(\frac{\mu}{\pi}(x)\right)\nabla \log\left(\frac{\mu}{\pi}(y)\right)k(x,y)d\mu(x)d\mu(y)\end{aligned}$$

+ I.P.P 3 times ($\nabla \log \mu(x) d\mu(x) = \nabla \mu(x)$) recovers the formula of the previous slide.

Stein identity and link with MMD

Under mild assumptions on k and π, the Stein kernel k_π is p.s.d. and satisfies a **Stein identity** [Oates et al., 2017]

$$\boxed{\int_{\mathbb{R}^d} k_\pi(x,.)d\pi(x) = 0.}$$

Consequently, **KSD is an MMD** with kernel k_π, since:

$$\begin{aligned}
\mathrm{MMD}^2(\mu|\pi) &= \int k_\pi(x,y)d\mu(x)d\mu(y) + \int k_\pi(x,y)d\pi(x)d\pi(y) \\
&\quad - 2\int k_\pi(x,y)d\mu(x)d\pi(y) \\
&= \int k_\pi(x,y)d\mu(x)d\mu(y) \\
&= \mathrm{KSD}^2(\mu|\pi)
\end{aligned}$$

KSD benefits

KSD can be computed when

- one has access to the score of π
- μ is a discrete measure, e.g. $\mu = \frac{1}{N}\sum_{i=1}^{N}\delta_{x^i}$, then :

$$\mathrm{KSD}^2(\mu|\pi) = \frac{1}{N^2}\sum_{i,j=1}^{N} k_{\pi}(x^i, x^j).$$

KSD benefits

KSD can be computed when

- one has access to the score of π
- μ is a discrete measure, e.g. $\mu = \frac{1}{N}\sum_{i=1}^{N} \delta_{x^i}$, then :

$$\mathrm{KSD}^2(\mu|\pi) = \frac{1}{N^2}\sum_{i,j=1}^{N} k_\pi(x^i, x^j).$$

KSD is known to metrize weak convergence [Gorham and Mackey, 2017] when:

- π is strongly log-concave at infinity ("distantly dissipative", e.g. true for gaussian mixtures)
- k has a slow decay rate, e.g. true when k is the IMQ kernel defined by $k(x,y) = (c^2 + \|x-y\|_2^2)^\beta$ for $c > 0$ and $\beta \in (-1, 0)$.

KSD in the literature

The KSD has been used for

- nonparametric statistical tests for goodness-of-fit [Xu and Matsuda, 2020, Kanagawa et al., 2020]
- sampling tasks:
  - (greedy algorithms) to select a suitable set of static points to approximate π, adding a new one at each iteration [Chen et al., 2018, Chen et al., 2019]
  - to compress [Riabiz et al., 2020] or reweight [Hodgkinson et al., 2020] Markov Chain Monte Carlo (MCMC) outputs
  - to learn a static transport map from μ_0 to π [Fisher et al., 2020].
  - learn Energy-Based models $\pi \propto \exp(-V)$ from samples of π (use reverse KSD) [Domingo-Enrich et al., 2021]

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

Time/Space discretization of the KSD gradient flow

Let $\mathcal{F}(\mu) = \mathrm{KSD}^2(\mu|\pi)$.

- Its Wasserstein gradient flow on $\mathcal{P}_2(\mathbb{R}^d)$ finds a continuous path of distributions that decreases $\mathcal{F}$.
- Different algorithms to approximate π depend on the time and space discretization of this flow.

Forward discretization: Wasserstein gradient descent

Discrete measures: For discrete measures $\hat{\mu} = \frac{1}{N}\sum_{i=1}^{N} \delta_{x^i}$, we have an explicit loss function

$$L([x^i]_{i=1}^N) := \mathcal{F}(\hat{\mu}) = \frac{1}{N^2}\sum_{i,j=1}^{N} k_\pi(x^i, x^j).$$

Then, Wasserstein gradient descent of $\mathcal{F}$ for discrete measures

$$\Updownarrow$$

(Euclidean) gradient descent of L on the particles.

KSD Descent - algorithms

We propose two ways to implement KSD Descent:

Algorithm 1 KSD Descent GD

Input: initial particles $(x_0^i)_{i=1}^N \sim \mu_0$, number of iterations M, step-size γ

for $n = 1$ **to** M **do**

$$[x_{n+1}^i]_{i=1}^N = [x_n^i]_{i=1}^N - \frac{2\gamma}{N^2} \sum_{j=1}^N [\nabla_2 k_\pi(x_n^j, x_n^i)]_{i=1}^N,$$

end for

Return: $[x_M^i]_{i=1}^N$.

Algorithm 2 KSD Descent L-BFGS

Input: initial particles $(x_0^i)_{i=1}^N \sim \mu_0$, tolerance tol

Return: $[x_*^i]_{i=1}^N = \text{L-BFGS}(L, \nabla L, [x_0^i]_{i=1}^N, \text{tol})$.

L-BFGS [Liu and Nocedal, 1989] is a quasi Newton algorithm that is faster and more robust than Gradient Descent, and **does not require the choice of step-size!**

L-BFGS

L-BFGS (Limited memory Broyden–Fletcher–Goldfarb–Shanno algorithm) is a quasi-Newton method:

$$x_{n+1} = x_n - \gamma_n B_n^{-1} \nabla L(x_n) := x_n + \gamma_n d_n \tag{4}$$

where B_n^{-1} is a p.s.d. matrix approximating the inverse Hessian at x_n.

Step1. (requires ∇L) It computes a cheap version of d_n based on BFGS recursion:

$$B_{n+1}^{-1} = \left(I - \frac{\Delta x_n y_n^T}{y_n^T \Delta x_n} \right) B_n^{-1} \left(I - \frac{y_n \Delta x_n^T}{y_n^T \Delta x_n} \right) + \frac{\Delta x_n \Delta x_n^T}{y_n^T \Delta x_n}$$

where

$$\begin{aligned} \Delta x_n &= x_{n+1} - x_n \\ y_n &= \nabla L(x_{n+1}) - \nabla L(x_n) \end{aligned}$$

Step2. (requires L and ∇L) A line-search is performed to find the best step-size in (4) :

$$\begin{aligned} L(x_n + \gamma_n d_n) &\leq L(x_n) + c_1 \gamma_n \nabla L(x_n)^T d_n \\ \nabla L(x_n + \gamma_n d_n)^T d_n &\geq c_2 \nabla L(x_n)^T d_n \end{aligned}$$

See [Nocedal and Wright, 2006].

Related work

1. minimize the **KL divergence** (requires $\nabla \log \pi$), e.g. with **Stein Variational Gradient descent** (SVGD, [Liu and Wang, 2016]).

Uses a set of N interacting particles and a p.s.d. kernel $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ to approximate π:

$$x_{n+1}^i = x_n^i - \gamma \left[\frac{1}{N} \sum_{j=1}^{N} k(x_n^i, x_n^j) \nabla \log \pi(x_n^j) + \nabla_1 k(x_n^j, x_n^i) \right],$$

Does not minimize a closed-form functional for discrete measures! $\Longrightarrow$ cannot use L-BFGS.

2. minimize the **MMD** [Arbel et al., 2019]

$$x_{n+1}^i = x_n^i - \gamma \left[\frac{1}{N} \sum_{j=1}^{N} \nabla_2 k(x_n^j, x_n^i) - \nabla_2 k(y^j, x_n^i) \right].$$

(requires samples $(y_j)_{j=1}^N \sim \pi$)

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

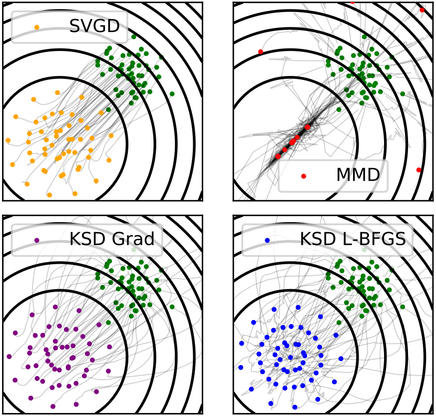
KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

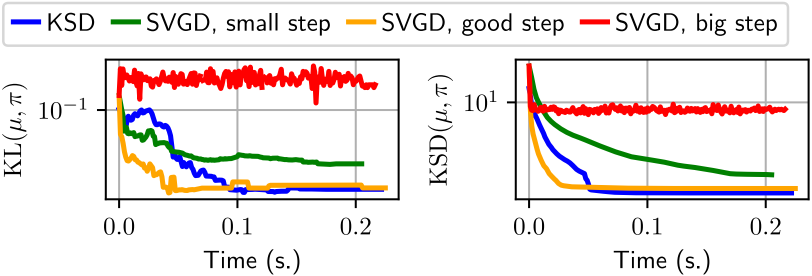
Toy experiments - 2D standard gaussian



The green points represent the initial positions of the particles.

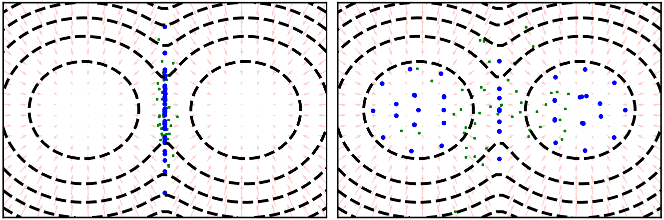
The light grey curves correspond to their trajectories.

SVGD vs KSD Descent - importance of the step-size



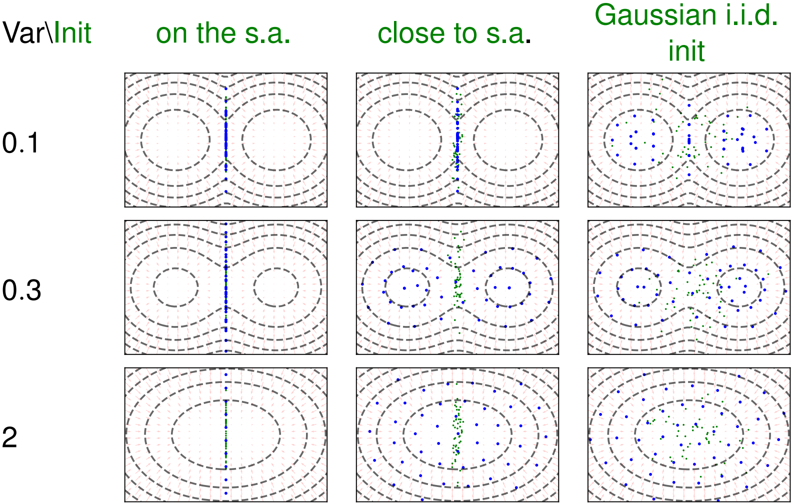
Convergence speed of KSD and SVGD on a Gaussian problem in 1D, with 30 particles.

2D mixture of (isolated) Gaussians - failure cases



The green crosses indicate the initial particle positions
the blue ones are the final positions
The light red arrows correspond to the score directions.

More initializations



Green crosses : initial particle positions
Blue crosses : final positions

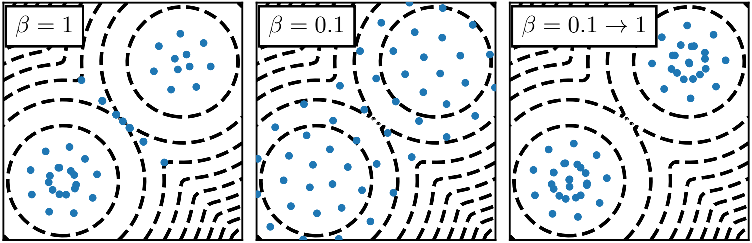
Stationary measures - some explanations

In the paper, we explain how particles can get stuck in planes of symmetry of the target π.

- we show that if a stationary measure μ_∞ is full support, then $\mathcal{F}(\mu_\infty) = 0$.
- but we also show that if *supp*$(\mu_0) \subset \mathcal{M}$, where $\mathcal{M}$ is a plane of symmetry of π, then for any time t it remains true for μ_t: *supp*$(\mu_t) \subset \mathcal{M}$.

Isolated Gaussian mixture - annealing

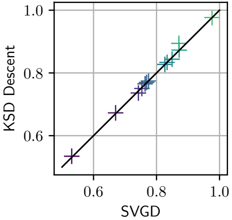
Add an inverse temperature variable β : $\pi^{\beta}(x) \propto \exp(-\beta V(x))$, with $0 < \beta \leq 1$ (i.e. multiply the score by β.)



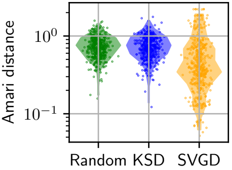
This is a hard problem, even for Langevin diffusions, where tempering strategies also have been proposed.

Beyond Log-concavity: Provable Guarantees for Sampling Multi-modal Distributions using Simulated Tempering Langevin Monte Carlo. Rong Ge, Holden Lee, Andrej Risteski. 2017.

Real world experiments (10 particles)

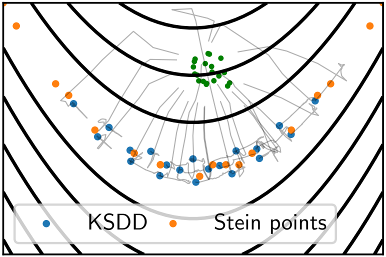


Bayesian logistic regression.
Accuracy of the KSD descent and SVGD for 13 datasets ($d \approx 50$).
Both methods yield similar results. KSD is better by 2% on one dataset.
Hint: convex likelihood.



Bayesian ICA.
Each dot is the Amari distance between an estimated matrix and the true unmixing matrix ($d \leq 8$).
KSD is not better than random.
Hint: highly non-convex likelihood.

So.. when does it work?



Comparison of KSD Descent and Stein points on a “banana” distribution. Green points are the initial points for KSD Descent. Both methods work successfully here, **even though it is not a log-concave distribution.**

We posit that KSD Descent succeeds because **there is no saddle point in the potential.**

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

First strategy : functional inequality?

$\mathcal{F}(\mu|\pi) = \iint k_\pi(x,y)d\mu(x)d\mu(y)$.

We have

$$\frac{\partial \mathcal{F}(\mu)}{\partial \mu} = \int k_\pi(x,.)d\mu(x) = \mathbb{E}_{x\sim\mu}[k_\pi(x,.)]$$

and under appropriate growth assumptions on k_π:

$$\nabla_{W_2}\mathcal{F}(\mu) = \mathbb{E}_{x\sim\mu}[\nabla_2 k_\pi(x,\cdot)],$$

Hence

$$\begin{aligned}\frac{d\mathcal{F}(\mu_t)}{dt} &= \langle \nabla_{W_2}\mathcal{F}(\mu_t), -\nabla_{W_2}\mathcal{F}(\mu_t)\rangle_{L^2(\mu_t)}\\ &= -\mathbb{E}_{y\sim\mu_t}\left[\|\mathbb{E}_{x\sim\mu_t}[\nabla_2 k_\pi(x,y)]\|^2\right] \leq 0.\end{aligned}$$

$\Longrightarrow$**Difficult to identify a functional inequality to relate** $d\mathcal{F}(\mu_t)/dt$ **to** $\mathcal{F}(\mu_t)$, and establish convergence in continuous time (similar to [Arbel et al., 2019]).

Second strategy : geodesic convexity of the KSD?

Let $\psi \in C_c^\infty(\mathbb{R}^d)$ and the path $\rho_t = (I + t\nabla\psi)_\#\mu$ for $t \in [0,1]$.

Define the quadratic form $\mathrm{Hess}_\mu \mathcal{F}(\psi,\psi) := \frac{d^2}{dt^2}\Big|_{t=0} \mathcal{F}(\rho_t)$, which is related to the W_2 **Hessian of $\mathcal{F}$ at μ.**

For $\psi \in C_c^\infty(\mathbb{R}^d)$, we have

$$\begin{aligned}\mathrm{Hess}_\mu \mathcal{F}(\psi,\psi) = \mathbb{E}_{x,y\sim\mu}&\left[\nabla\psi(x)^T \nabla_1\nabla_2 k_\pi(x,y)\nabla\psi(y)\right]\\ &+ \mathbb{E}_{x,y\sim\mu}\left[\nabla\psi(x)^T H_1 k_\pi(x,y)\nabla\psi(x)\right].\end{aligned}$$

The first term is always positive but not the second one.

$\Longrightarrow$ **the KSD is not convex w.r.t. W_2 geodesics.**

Third strategy : curvature near equilibrium?

What happens near equilibrium π? the second term vanishes due to the Stein property of k_π and :

$$\mathrm{Hess}_\pi \mathcal{F}(\psi,\psi) = \|S_{\pi,k_\pi}\mathcal{L}_\pi\psi\|^2_{\mathcal{H}_{k_\pi}} \geq 0$$

where

$$\mathcal{L}_\pi : f \mapsto -\Delta f - \langle \nabla \log \pi, \nabla f\rangle_{\mathbb{R}^d}$$

$$S_{\mu,k_\pi} : f \mapsto \int k_\pi(x,.)f(x)d\mu(x) \in \mathcal{H}_{k_\pi} = \overline{\left\{k_\pi(x,.), x \in \mathbb{R}^d\right\}}$$

Question: can we bound from below the Hessian at π by a quadratic form on the tangent space of $\mathcal{P}_2(\mathbb{R}^d)$ at π ($\subset L^2(\pi)$)?

$$\boxed{\|S_{\pi,k_\pi}\mathcal{L}_\pi\psi\|^2_{\mathcal{H}_{k_\pi}} = \mathrm{Hess}_\pi \mathcal{F}(\psi,\psi) \geq \lambda \|\nabla\psi\|^2_{L^2(\pi)} \text{ ?}}$$

That would imply exponential decay of $\mathcal{F}$ near π.

Curvature near equilibrium - negative result

The previous inequality

$$\boxed{\|S_{\pi,k_\pi}\mathcal{L}_\pi\psi\|^2_{\mathcal{H}_{k_\pi}} \geq \lambda\|\nabla\psi\|^2_{L^2(\pi)}}$$

- can be seen as a kernelized version of the Poincaré inequality for π :
$$\|\mathcal{L}_\pi\psi\|^2_{L_2(\pi)} \geq \lambda_\pi\|\nabla\psi\|^2_{L_2(\pi)}.$$
- can be written:
$$\langle\psi, P_{\pi,k_\pi}\psi\rangle_{L_2(\pi)} \geq \lambda\langle\psi, \mathcal{L}_\pi^{-1}\psi\rangle_{L_2(\pi)},$$
where $P_{\pi,k_\pi} : L^2(\pi) \to L^2(\pi), f \mapsto \int k_\pi(x,.)f(x)d\pi(x).$

Theorem : Let $\pi \propto e^{-V}$. Assume that $V \in C^2(\mathbb{R}^d)$, ∇V is Lipschitz and $\mathcal{L}_\pi$ has discrete spectrum. Then exponential decay near equilibium does not hold.

Outline

Introduction

Part I : Sampling as optimization of the KL

SVGD algorithm

Some non-asymptotic results

Part II : Sampling as optimization of the KSD

Preliminaries on Kernel Stein Discrepancy

KSD Descent

Experiments

Theoretical properties of the KSD flow

Conclusion

Conclusion

Pros:

- KSD Descent is a very simple algorithm, and can be used with L-BFGS [Liu and Nocedal, 1989] (fast, and does not require the choice of a step-size as in SVGD)
- works well on log-concave targets (unimodal gaussian, Bayesian logistic regression with gaussian priors) or "nice" distributions (banana)

Cons:

- KSD is not convex w.r.t. W_2, and no exponential decay near equilibrium holds
- does not work well on non log-concave targets (mixture of isolated gaussians, Bayesian ICA)

Open questions

- explain the convergence of KSD Descent when π is log-concave?
- quantify propagation of chaos ? (KSD for a finite number of particles vs infinite - but non uniformly Lipschitz vector field)
- how good is KSD quantisation?

Code

- Python package to try KSD descent yourself:
  pip install ksddescent
- website: `pierreablin.github.io/ksddescent/`
- It also features pytorch/numpy code for SVGD.

```
>>> import torch
>>> from ksddescent import ksdd_lbfgs
>>> n, p = 50, 2
>>> x0 = torch.rand(n, p)  # start from uniform distribution
>>> score = lambda x: x  # simple score function
>>> x = ksdd_lbfgs(x0, score)  # run the algorithm
```

Thank you for listening!

References I

Alquier, P. and Ridgway, J. (2017).
Concentration of tempered posteriors and of their variational approximations.
arXiv preprint arXiv:1706.09293.

Ambrosio, L., Gigli, N., and Savaré, G. (2008).
Gradient flows: in metric spaces and in the space of probability measures.
Springer Science & Business Media.

Arbel, M., Korba, A., Salim, A., and Gretton, A. (2019).
Maximum mean discrepancy gradient flow.
In *Advances in Neural Information Processing Systems*, pages 6481–6491.

References II

Carrillo, J. A., Craig, K., and Patacchini, F. S. (2019).
A blob method for diffusion.
Calculus of Variations and Partial Differential Equations, 58(2):1–53.

Chen, W. Y., Barp, A., Briol, F.-X., Gorham, J., Girolami, M., Mackey, L., Oates, C., et al. (2019).
Stein point Markov Chain Monte Carlo.
Proceedings of the 36th International Conference on Machine Learning,.

Chen, W. Y., Mackey, L., Gorham, J., Briol, F.-X., and Oates, C. J. (2018).
Stein points.
Proceedings of the 35th International Conference on Machine Learning,.

References III

Chu, C., Minami, K., and Fukumizu, K. (2020).
The equivalence between stein variational gradient descent and black-box variational inference.
arXiv preprint arXiv:2004.01822.

Chwialkowski, K., Strathmann, H., and Gretton, A. (2016).
A kernel test of goodness of fit.
In *International conference on machine learning*.

Dalalyan, A. S. (2017).
Further and stronger analogy between sampling and optimization: Langevin monte carlo and gradient descent.
arXiv preprint arXiv:1704.04752.

References IV

Detommaso, G., Cui, T., Marzouk, Y., Spantini, A., and Scheichl, R. (2018).
A stein variational newton method.
In *Advances in Neural Information Processing Systems*, pages 9169–9179.

Domingo-Enrich, C., Bietti, A., Vanden-Eijnden, E., and Bruna, J. (2021).
On energy-based models with overparametrized shallow neural networks.
arXiv preprint arXiv:2104.07531.

Duncan, A., Nüsken, N., and Szpruch, L. (2019).
On the geometry of stein variational gradient descent.
arXiv preprint arXiv:1912.00894.

References V

Durmus, A., Majewski, S., and Miasojedow, B. (2019).
Analysis of langevin monte carlo via convex optimization.
Journal of Machine Learning Research, 20(73):1–46.

Durmus, A. and Moulines, E. (2016).
Sampling from strongly log-concave distributions with the unadjusted langevin algorithm.
arXiv preprint arXiv:1605.01559, 5.

Feng, Y., Wang, D., and Liu, Q. (2017).
Learning to draw samples with amortized stein variational gradient descent.
arXiv preprint arXiv:1707.06626.

References VI

Fisher, M. A., Nolan, T., Graham, M. M., Prangle, D., and Oates, C. J. (2020).
Measure transport with kernel Stein discrepancy.
arXiv preprint arXiv:2010.11779.

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

Hodgkinson, L., Salomone, R., and Roosta, F. (2020).
The reproducing Stein kernel approach for post-hoc corrected sampling.
arXiv preprint arXiv:2001.09266.

References VII

- Jordan, R., Kinderlehrer, D., and Otto, F. (1998).
  The variational formulation of the fokker–planck equation.
  SIAM journal on mathematical analysis, 29(1):1–17.

- Kanagawa, H., Jitkrittum, W., Mackey, L., Fukumizu, K., and Gretton, A. (2020).
  A kernel Stein test for comparing latent variable models.
  arXiv preprint arXiv:1907.00586.

- Karimi, H., Nutini, J., and Schmidt, M. (2016).
  Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition.
  In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer.

References VIII

Liu, C. and Zhu, J. (2018).
Riemannian stein variational gradient descent for bayesian inference.
In *Thirty-second aaai conference on artificial intelligence*.

Liu, D. C. and Nocedal, J. (1989).
On the limited memory BFGS method for large scale optimization.
Mathematical programming, 45(1-3):503–528.

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

References IX

Liu, Q., Lee, J., and Jordan, M. (2016).
A kernelized stein discrepancy for goodness-of-fit tests.
In *International conference on machine learning*, pages 276–284.

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

Liu, Y., Ramachandran, P., Liu, Q., and Peng, J. (2017).
Stein variational policy gradient.
arXiv preprint arXiv:1704.02399.

References X

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

Nocedal, J. and Wright, S. (2006).
Numerical optimization.
Springer Science & Business Media.

Oates, C. J., Girolami, M., and Chopin, N. (2017).
Control functionals for monte carlo integration.
Journal of the Royal Statistical Society: Series B (Statistical Methodology), 79(3):695–718.

References XI

Otto, F. (2001).
The Geometry of Dissipative Evolution Equations: The Porous Medium Equation.
Communications in Partial Differential Equations, 26(1-2):101–174.

Pu, Y., Gan, Z., Henao, R., Li, C., Han, S., and Carin, L. (2017).
Vae learning via stein variational gradient descent.
In *Advances in Neural Information Processing Systems*, pages 4236–4245.

Riabiz, M., Chen, W., Cockayne, J., Swietach, P., Niederer, S. A., Mackey, L., Oates, C., et al. (2020).
Optimal thinning of MCMC output.
arXiv preprint arXiv:2005.03952.

References XII

Steinwart, I. and Christmann, A. (2008).
Support vector machines.
Springer Science & Business Media.

Wang, D. and Liu, Q. (2016).
Learning to draw samples: With application to amortized mle for generative adversarial learning.
arXiv preprint arXiv:1611.01722.

Wibisono, A. (2018).
Sampling as optimization in the space of measures: The langevin dynamics as a composite optimization problem.
arXiv preprint arXiv:1802.08089.

References XIII

Xu, W. and Matsuda, T. (2020).
A Stein goodness-of-fit test for directional distributions.
In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, volume 108, pages 320–330. PMLR.

Zhang, C., Butepage, J., Kjellstrom, H., and Mandt, S. (2018).
Advances in variational inference.
IEEE transactions on pattern analysis and machine intelligence.

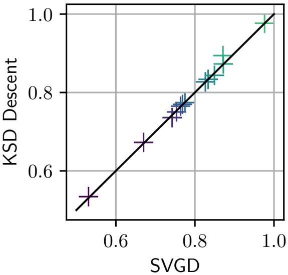
1 - Bayesian Logistic regression

Datapoints $d_1, \dots, d_q \in \mathbb{R}^p$, and labels $y_1, \dots, y_q \in \{\pm 1\}$.

Labels y_i are modelled as $p(y_i = 1 | d_i, w) = (1 + \exp(-w^\top d_i))^{-1}$ for some $w \in \mathbb{R}^p$.

The parameters w follow the law $p(w|\alpha) = \mathcal{N}(0, \alpha^{-1} I_p)$, and $\alpha > 0$ is drawn from an exponential law $p(\alpha) = \mathrm{Exp}(0.01)$.

The parameter vector is then $x = [w, \log(\alpha)] \in \mathbb{R}^{p+1}$, and we use KSD-LBFGS to obtain samples from $p(x| \left(d_i, y_i\right)_{i=1}^q)$ for 13 datasets, with $N = 10$ particles for each.



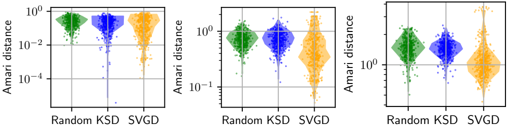
Accuracy of the KSD descent and SVGD on bayesian logistic regression for 13 datasets.
Both methods yield similar results. KSD is better by 2% on one dataset.

2 - Bayesian Independent Component Analysis

ICA: $x = W^{-1}s$, where x is an observed sample in $\mathbb{R}^p$, $W \in \mathbb{R}^{p\times p}$ is the unknown square unmixing matrix, and $s \in \mathbb{R}^p$ are the independent sources.

1)Assume that each component has the same density $s_i \sim p_s$.
2) The likelihood of the model is $p(x|W) = \log|W| + \sum_{i=1}^{p} p_s([Wx]_i)$.
3)Prior: W has i.i.d. entries, of law $\mathcal{N}(0, 1)$.

The posterior is $p(W|x) \propto p(x|W)p(W)$, and the score is given by $s(W) = W^{-\top} - \psi(Wx)x^\top - W$, where $\psi = -\frac{p_s'}{p_s}$. In practice, we choose p_s such that $\psi(\cdot) = \mathtt{tanh}(\cdot)$. We then use the presented algorithms to draw 10 particles $W \sim p(W|x)$ on 50 experiments.



Left: $p = 2$. Middle: $p = 4$. Right: $p = 8$.
Each dot = Amari distance between an estimated matrix and the true unmixing matrix.
KSD Descent is not better than random. Explanation: ICA likelihood is highly non-convex.