


Understanding and Accelerating Particle-Based Variational Inference

Chang Liu[†], Jingwei Zhuo[†], Pengyu Cheng[‡], Ruiyi Zhang[‡],
Jun Zhu[†§], Lawrence Carin[‡§]

ICML 2019

† Tsinghua University ‡ Duke University
§: Corresponding authors

Introduction

Particle-based Variational Inference Methods (ParVIs):

- Represent the variational distribution q by particles; update the particles to minimize $\mathrm{KL}_p(q)$.
- More flexible than classical VIs; more particle-efficient than MCMCs.

Related Work:

- Stein Variational Gradient Descent (SVGD) [3] simulates the gradient flow (steepest descending curves) of KL_p on $\mathcal{P}_{\mathcal{H}}(\mathcal{X})$ [2].
- The Blob and DGF methods [1] simulate the gradient flow of KL_p on the Wasserstein space $\mathcal{P}_2(\mathcal{X})$.

ParVIs Approximate $\mathcal{P}_2(\mathcal{X})$ (Wasserstein) Gradient Flow

Remark 1

Existing ParVI methods approximate Wasserstein Gradient flow by smoothing the **density** *or* **functions**.

Smoothing the Density

- Blob [1] partially smooths the density.

$$v^{\text{GF}} = -\nabla\big(\frac{\delta}{\delta q}\mathbb{E}_q[\log(q/p)]\big) \Longrightarrow v^{\text{Blob}} = -\nabla\big(\frac{\delta}{\delta q}\mathbb{E}_q[\log(\tilde{q}/p)]\big).$$

- **GFSD** fully smooths the density.

$$v^{\text{GF}} := \nabla \log p - \nabla \log q \Longrightarrow v^{\text{GFSD}} := \nabla \log p - \nabla \log \tilde{q}.$$

Smoothing Functions

- SVGD restricts the optimization domain $\mathcal{L}^2_q$ to $\mathcal{H}^D$.
- **GFSF** smoothed functions in a similar way: $\hat{v}^{\text{GFSF}} = \hat{g} + \hat{K}'\hat{K}^{-1}$. (Note $\hat{v}^{\text{SVGD}} = \hat{v}^{\text{GFSF}}\hat{K}$.)

$\hat{g}_{:,i} = \nabla_{x^{(i)}} \log p(x^{(i)})$, $\hat{K}_{ij} = K(x^{(i)}, x^{(j)})$, $\hat{K}'_{:,i} = \sum_j \nabla_{x^{(j)}} K(x^{(j)}, x^{(i)})$.

ParVIs Approximate $\mathcal{P}_2(\mathcal{X})$ Gradient Flow by Smoothing

- Equivalence:
  Smoothing-function objective $= \mathbb{E}_q[L(v)]$, $L : \mathcal{L}^2_q \to L^2_q$ linear.
  $$\Longrightarrow \mathbb{E}_{\tilde{q}}[L(v)] = \mathbb{E}_{q*K}[L(v)] = \mathbb{E}_q[L(v) * K] = \mathbb{E}_q[L(v * K)].$$
- Necessity: $\operatorname{grad} \mathrm{KL}_p(q)$ undefined at $q = \hat{q} := \frac{1}{N}\sum_{i=1}^N \delta_{x^{(i)}}$.

Theorem 2 (Necessity of smoothing for SVGD)

For $q = \hat{q}$ and $v \in \mathcal{L}^2_p$:

$$\max_{v \in \mathcal{L}^2_p, \|v\|_{\mathcal{L}^2_p} = 1} \left\langle v^{\mathrm{GF}}, v \right\rangle_{\mathcal{L}^2_{\hat{q}}},$$

has no optimal solution.

ParVIs rely on the smoothing assumption!
No free lunch!

Bandwidth Selection via the Heat Equation

Note

Under the dynamics $\mathrm{d}x = -\nabla \log q_t(x)\,\mathrm{d}t$, q_t evolves following the heat equation (HE): $\partial_t q_t(x) = \Delta q_t(x)$.

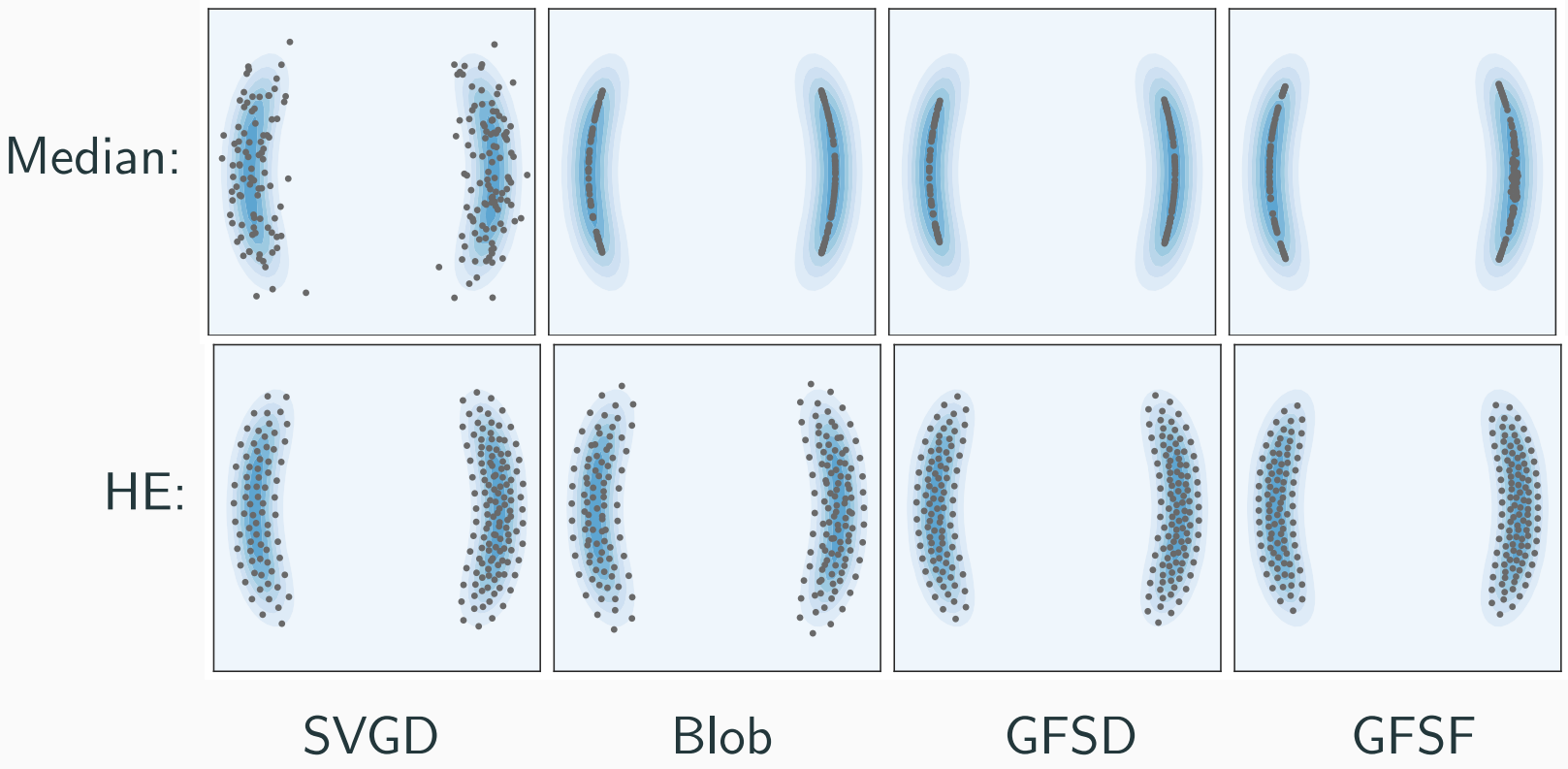

Figure 1: Comparison of HE (bottom row) with the median method (top row) for bandwidth selection.

Nesterov's Acceleration Method on Riemannian Manifolds

- Riemannian Accelerated Gradient (RAG) [4] (with simplification):

$$\begin{cases} q_k = \mathrm{Exp}_{r_{k-1}}(\varepsilon v_{k-1}), \\ r_k = \mathrm{Exp}_{q_k}\left[-\Gamma_{r_{k-1}}^{q_k}\left(\frac{k-1}{k}\mathrm{Exp}_{r_{k-1}}^{-1}(q_{k-1}) - \frac{k+\alpha-2}{k}\varepsilon v_{k-1}\right)\right]. \end{cases}$$

- Riemannian Nesterov's method (RNes) [5] (with simplification):

$$\begin{cases} q_k = \mathrm{Exp}_{r_{k-1}}(\varepsilon v_{k-1}), \\ r_k = \mathrm{Exp}_{q_k}\left\{c_1 \mathrm{Exp}_{q_k}^{-1}\left[\mathrm{Exp}_{r_{k-1}}\left((1-c_2)\mathrm{Exp}_{r_{k-1}}^{-1}(q_{k-1}) + c_2 \mathrm{Exp}_{r_{k-1}}^{-1}(q_k)\right)\right]\right\}. \end{cases}$$

- Inverse exponential map: computationally expensive

Proposition 3 (Inverse exponential map)
For pairwise close samples $\{x^{(i)}\}_i$ *of* q *and* $\{y^{(i)}\}_i$ *of* r*, we have* $\left(\mathrm{Exp}_q^{-1}(r)\right)(x^{(i)}) \approx y^{(i)} - x^{(i)}$.

- Parallel transport: hard to implement

Proposition 4 (Parallel transport)
For pairwise close samples $\{x^{(i)}\}_i$ *of* q *and* $\{y^{(i)}\}_i$ *of* r*, we have* $\left(\Gamma_q^r(v)\right)(y^{(i)}) \approx v(x^{(i)})$, $\forall v \in T_q\mathcal{P}_2$.

Acceleration Framework for ParVIs

Algorithm 1 The acceleration framework with Wasserstein Accelerated Gradient (WAG) and Wasserstein Nesterov's method (WNes)

WAG: select acceleration factor $\alpha > 3$;
WNes: select or calculate $c_1, c_2 \in \mathbb{R}^+$;
Initialize $\{x_0^{(i)}\}_{i=1}^N$ distinctly; let $y_0^{(i)} = x_0^{(i)}$;
for $k = 1, 2, \cdots, k_{\max}$, **do**
  for $i = 1, \cdots, N$, **do**
    Find $v(y_{k-1}^{(i)})$ by SVGD/Blob/DGF/GFSD/GFSF;
    $x_k^{(i)} = y_{k-1}^{(i)} + \varepsilon v(y_{k-1}^{(i)})$;
    $y_k^{(i)} = x_k^{(i)} + \begin{cases} \text{WAG: } \frac{k-1}{k}(y_{k-1}^{(i)} - x_{k-1}^{(i)}) + \frac{k+\alpha-2}{k}\varepsilon v(y_{k-1}^{(i)}); \\ \text{WNes: } c_1(c_2-1)(x_k^{(i)} - x_{k-1}^{(i)}); \end{cases}$
  end for
end for
Return $\{x_{k_{\max}}^{(i)}\}_{i=1}^N$.

Bayesian Logistic Regression (BLR)

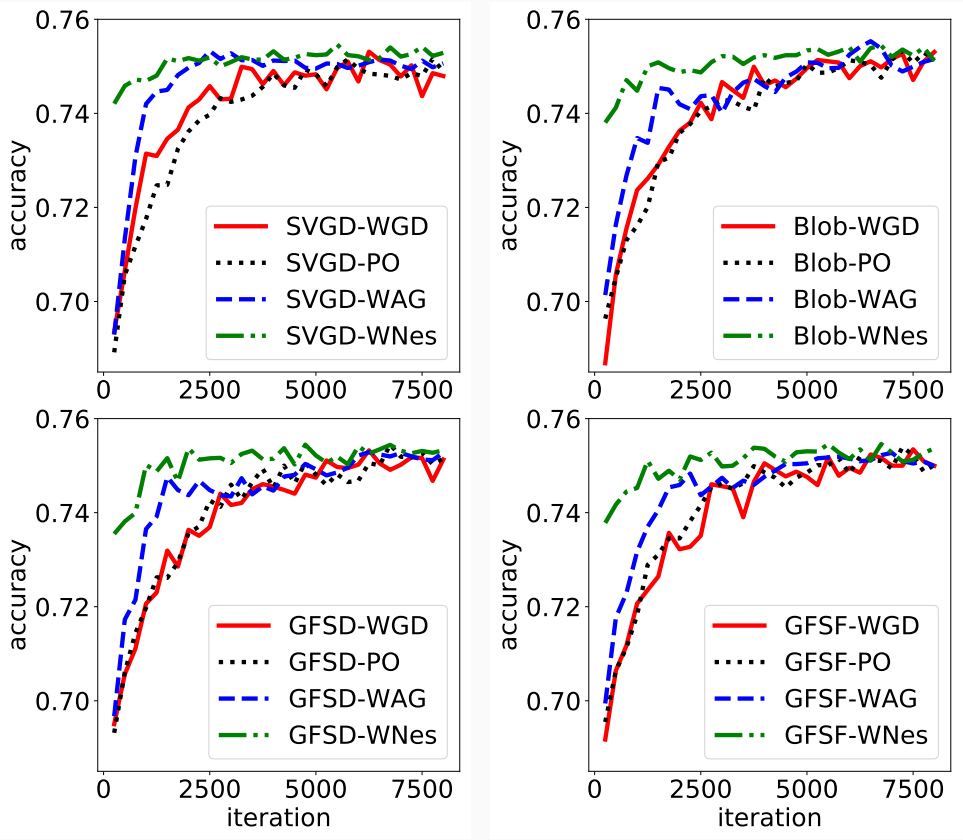

Figure 2: Acceleration effect of WAG and WNes on BLR on the Covertype dataset, measured by prediction accuracy on test dataset. Each curve is averaged over 10 runs.

Latent Dirichlet Allocation (LDA)

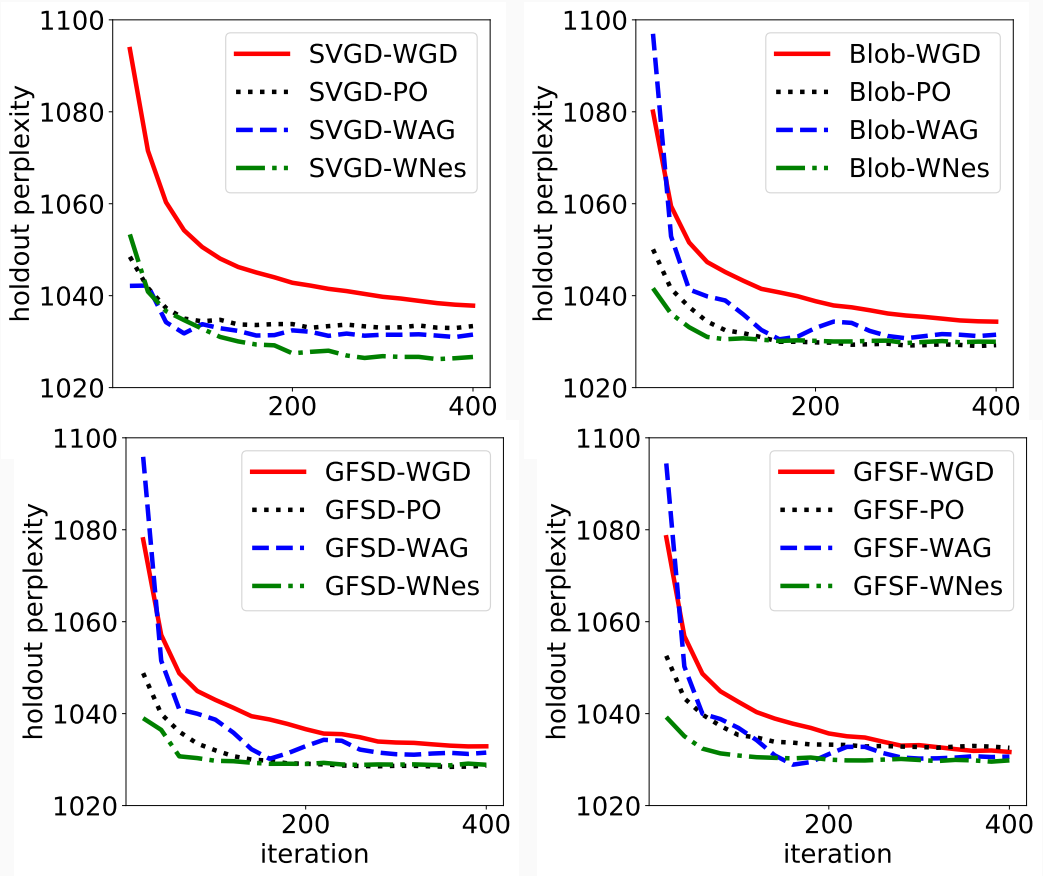

Figure 3: Acceleration effect of WAG and WNes on LDA. Inference results are measured by the hold-out perplexity. Curves are averaged over 10 runs.

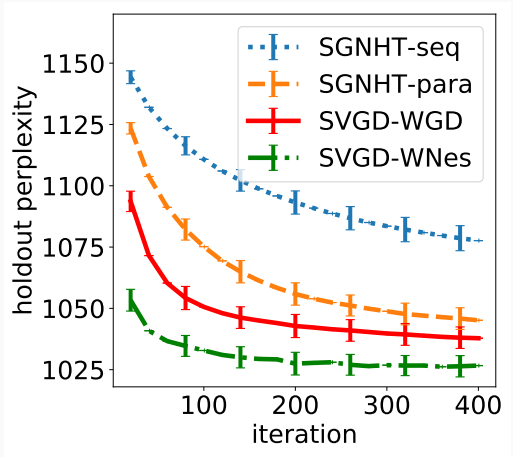

Figure 4: Comparison of SVGD and SGNHT on LDA, as representatives of ParVIs and MCMCs. Average over 10 runs.

Summary

Contributions (in theory):

- ParVIs approximate the Wasserstein gradient flow by a compulsory smoothing assumption.
- ParVIs either smooth the density or smooth functions, and they are equivalent.

Contributions (in practice):

- Two new ParVIs (GFSF and GFSD).
- A principled bandwidth selection method for the smoothing kernel.
- An acceleration framework for general ParVIs.

Changyou Chen, Ruiyi Zhang, Wenlin Wang, Bai Li, and Liqun Chen.

A unified particle-optimization framework for scalable bayesian sampling.

arXiv preprint arXiv:1805.11659, 2018.

Qiang Liu.

Stein variational gradient descent as gradient flow.

In *Advances in neural information processing systems*, pages 3118–3126, 2017.

Qiang Liu and Dilin Wang.

Stein variational gradient descent: A general purpose bayesian inference algorithm.

In *Advances In Neural Information Processing Systems*, pages 2378–2386, 2016.

Yuanyuan Liu, Fanhua Shang, James Cheng, Hong Cheng, and Licheng Jiao.

Accelerated first-order methods for geodesically convex optimization on riemannian manifolds.

In *Advances in Neural Information Processing Systems*, pages 4875–4884, 2017.

Hongyi Zhang and Suvrit Sra.

An estimate sequence for geodesically convex optimization.

In *Conference On Learning Theory*, pages 1703–1723, 2018.