LSVI: On-line supplement

Anonymous

March 31, 2025

Abstract

This document reports additional experimental results conducted in response to specific comments from the referees. We investigate the use of subsampling within LSVI to address scalability concerns, compare the performance of LSVI with natural gradient descent (NGD), analyze variability across repeated runs, and report misclassification rates in a logistic regression setting. Furthermore, we provide an empirical evaluation of the computational efficiency of LSVI in terms of wall-clock time compared to existing implementations, including pyMC3's ADVI and Blackjax's mean-field variational inference. We are grateful to the referees for asking those additional experiments. They give further evidence that LSVI is competitive with existing methods.

1 Subsampling

Following Ref. 8cK4's suggestion and Ref. W5YG's point about scalability, we report here extra results on using subsampling within LSVI.

Assuming that the log target density may be written as a sum over n terms as follows:

$$\log \pi(x) = \sum_{i=0}^{n} f_i(x)$$

one may derive an unbiased mini-batch estimate of $\log \pi$ by replacing $\log \pi$ with

$$\widehat{\log \pi}(x) = \frac{n}{n_{\text{batch}}} \sum_{i \in \mathcal{S}} f_i(x)$$

where the random set $\mathcal{S} \subset \{1, \ldots, n\}$ is obtained by sampling with replacement. This covers the standard Bayesian case where $\pi(x) \propto \pi_0(x) p_x(y_1, \ldots, y_n)$, where $y_1, \ldots, y_n$ are i.i.d observations $y_i \sim p_x(\mathrm{d}y)$.

We implement LSVI with $\log \pi(x)$ replaced by its mini-batch (stochastic) approximation to showcase that LSVI can be applied to big data scenarios, i.e. n is large. We consider the same settings in our first experiment in the paper (logistic regression, Gaussian prior, predictors are pre-processed in the same way), and apply LSVI to the Census dataset https://archive.ics.uci.edu/dataset/20/census+income), where $n \approx 49,000$. We set $n_{\text{batch}} = 1000$. A new batch is drawn at each iteration.

Figure 1 reports the KL loss (up to an unknown constant) across iterations (x-axis) and over repeated runs (red lines) for two different schedules. As expected, using a mini-batch approximation makes the results more noisy, but still converges quickly relative to the number of epochs (where an epoch is a block of k successive iterations, with $k = n/n_{\text{batch}}$).

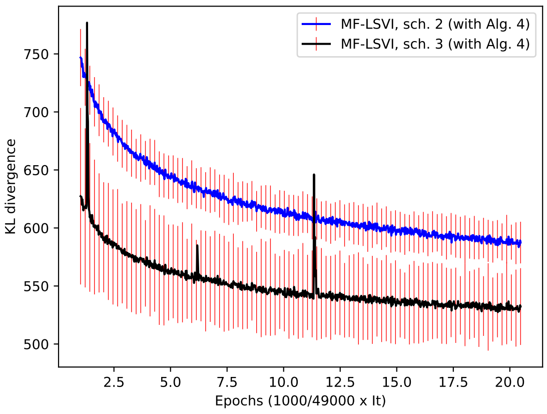
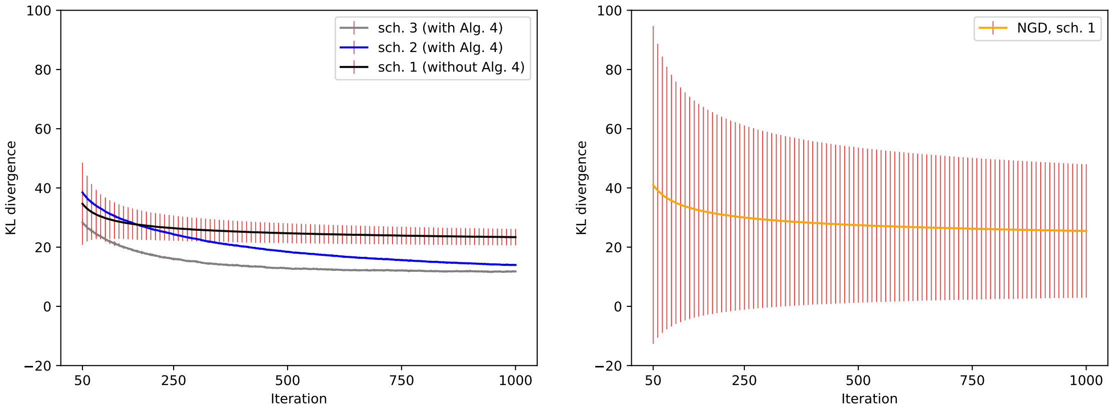


Figure 1: Logistic regression posterior with subsampling, Census-income dataset, diagonal (mean-field) covariance approximation, for MF-LSVI with schedules 2 and 3 (see supplement for details on schedules). Mean and one-standard deviation over 100 trials, $N = 10^4$ samples. Truncated from iteration 50 for better readability.

2 Comparison with NGD (natural gradient descent)

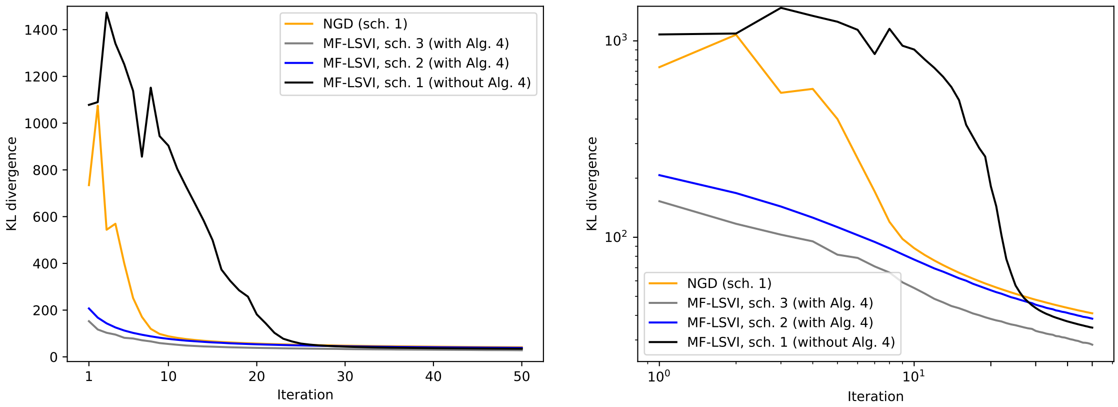
As requested by Refs. W5YG and 8cK4, we report here the performance of NGD in the same settings as in our logistic regression example; see Figures 2 and 3. The loss is evaluated via Monte-Carlo sampling, and the gradient is obtained via jax.grad. The Fisher matrix is evaluated via Monte-Carlo. The total cost per iteration is $O(d^3)$ for mean-field Gaussian distributions. The step sizes for both NGD and MF-LSVI (sch. 1) are linearly decreasing with t. See Table 3 in the manuscript for a description of the schedules.



(a) MF-LSVI (Alg. 2)

(b) NGD, schedule 1.

Figure 2: Logistic regression posterior, Sonar data, diagonal (mean-field) covariance approximation. Comparison of MF-LSVI (Alg. 2, with different schedules, left panel), with NGD (Schedule 1, right panel). Mean and one-standard deviation over 100 trials, $N = 10^4$ samples. Truncated from iteration 50 for better readability.



(a) Average loss (over 100 runs) as a function of time.

(b) Same plot on log-log scale

Figure 3: Logistic regression posterior, Sonar data, diagonal (mean-field) covariance approximation. Comparison of MF-LSVI (Alg. 2, with different schedules) with NGD. Mean 100 trials, $N = 10^4$ samples.

Our NGD implementation exhibits higher variance compared to MF-LSVI. A fair comparison for the stochastic NGD is MF-LSVI with schedule 1. since both are estimates of NGD with step sizes $\epsilon_t = 1/(t+1)$.

We left for future work a comparison with a fast implementation of stochastic scheme built upon NGD in which the Fisher matrix is updated using a single draw from the current Gaussian approximation. This allows for fast inversion of the new estimate of the Fisher matrix via Woodbury's identity.

3 Variability over repeated runs

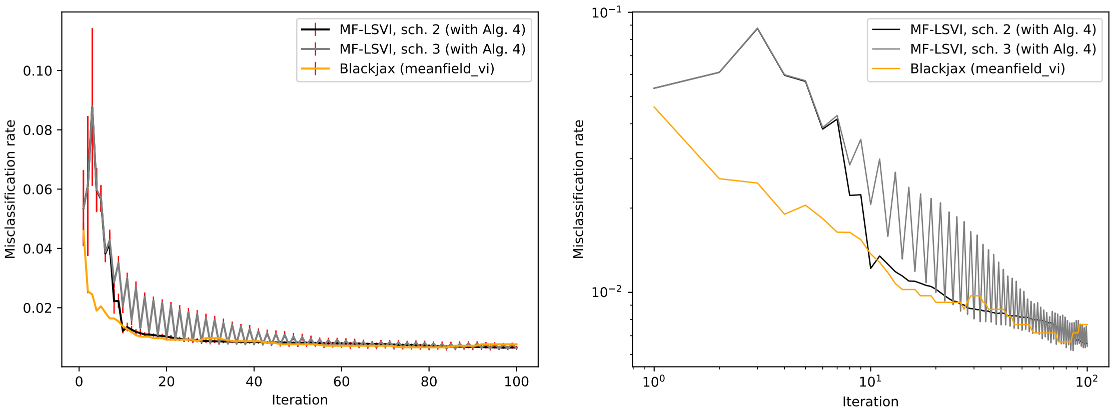
As recommended by Ref. hBY5, we now plot (e.g. Figures 1 and 2) the average loss obtained over 100 trials with a one-standard deviation interval for the mean-field case with Sonar dataset in the exact same setting as Figure 2 in the manuscript. We plan on replacing Figures 1, 2 and 5 in the paper with similar plots. Thanks to GPU parallelization in JAX, running multiple trials incurs little additional computational cost.

4 Reporting misclassification rates

Following hBY5's comment on misclassification rate, Fig. 4 reports this rate across iterations in the MNIST logistic regression problem.

The experiment is ran over 20 independent runs to obtain confidence intervals. The predictors are taken to be the means of the variational approximations.

All three descent methods (blackjax.meanfield_vi with step-size 10^{-3}, MF-LSVI with two different schedules, see Table 3) yield a low classification rate over the MNIST test dataset ($\approx 10^{-2}$). After a few dozen of iterations, no difference between classification rates can be observed.



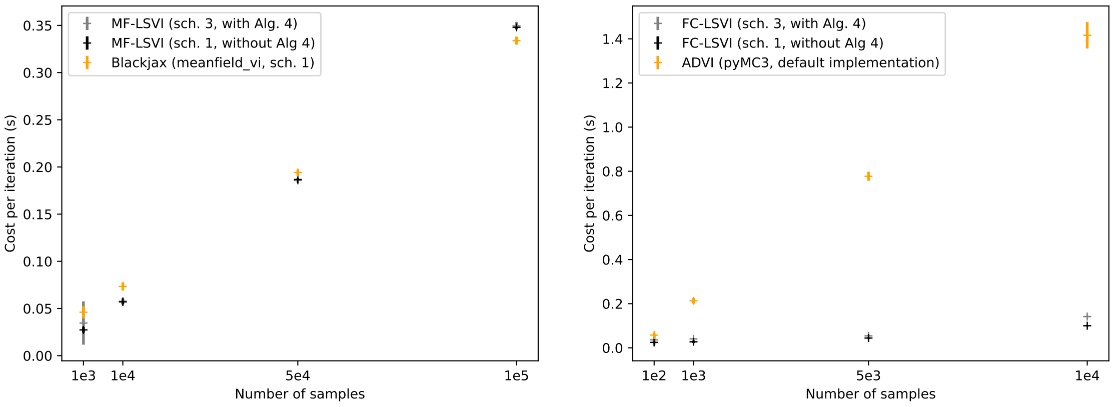
(a) Mean and one std interval over 20 trials.

(b) Mean over 20 trials, log-log scale.

Figure 4: Misclassification rate for the MNIST dataset. Diagonal covariance approximation, MF-LSVI (Alg. 2) vs Blackjax mean-field VI, $N = 10^4$ samples. Truncated to iteration 100 for better readability.

5 Wall-clock time

Following hBY5's comments, we conducted additional experiments to monitor the actual cost per iteration (in seconds) of our proposed methods compared to standard implementations (that is, the default pyMC3 implementation of ADVI and Blackjax implementation, blackjax.meanfield_vi). We found our method to have costs similar to those of the existing implementations. See Fig. 5.



(a) Mean-field approximation (MF-LSVI vs blackjax.meanfield_vi), MNIST dataset.

(b) Full-covariance approximation (FC-LSVI vs ADVI, pyMC), Sonar dataset.

Figure 5: Logistic regression posterior. Wall-clock time per iteration. Means and one-standard deviation intervals, over 5 trials, and 100 iterations.

We believe that computational complexity with respect to both N and d should be the primary criterion for comparing algorithms from a computational perspective, as the observed cost per iteration is heavily influenced by implementation details (e.g., the use of JAX versus PyTensor). We put emphasis on this point in the newer version of the manuscript.

6 References

Because of the length limitation, we list the full references to the cited articles in our answers to R3 and R4 below.

6.1 References (R3)

[1] M. Khan, et al. Conjugate-Computation Variational Inference : Converting Variational Inference in Non-Conjugate Models to Inferences in Conjugate Models. PMLR, 2017.

[2] M. Khan, et al. Fast and scalable Bayesian deep learning by weight-perturbation in ADAM. ICML, 2018.

[3] K. Wu, et al. Understanding stochastic natural gradient variational inference. ICML, 2024.

[4] G. P. Dehaene. Expectation propagation performs a smoothed gradient descent. arxiv, 2016.

[5] P.-C. Aubin-Frankowski, et al. Mirror descent with relative smoothness in measure spaces, with application to sinkhorn and EM. NIPS, 2024.

[6] H. Lu, et al. Relatively smooth convex optimization by first-order methods, and applications. SIOPT, 2018.

[7] M. E. Khan, et al. Kullback-leibler proximal variational inference. NIPS, 2015.

[8] M. J. Wainwright et al. Graphical models, exponential families, and variational inference. FTML, 2008.

[9] Salimans, T. et al. Fixed-Form Variational Posterior Approximation through Stochastic Linear Regression. Bayesian Analysis, 2013.

[10] W. Lin, et al. Fast and simple natural-gradient variational inference with mixture of exponential-family approximations. PMLR, 2019.

[11] Removed.

[12] J. Domke, et al. Provable convergence guarantees for black-box variational inference. NIPS, 2023.

6.2 References (R4)

[1] M. Welandawe, et al. A framework for improving the reliability of black-box variational inference. JMLR, 2024.

[2] K. Scaman, et al. Robustness analysis of non-convex stochastic gradient descent using biased expectations. NeurIPS, 2020.

[3] K. Scaman, et al. Convergence rates of non-convex stochastic gradient descent under a generic lojasiewicz condition and local smoothness. ICML, 2022.

[4] B. Batardi‘ere, et al. Importance sampling-based gradient method for dimension reduction in poisson log-normal model, 2024.

[5] R. Y. Chen, et al. The masked sample covariance estimator: an analysis using matrix concentration inequalities. Information and Inference: A Journal of the IMA, 2012.

[6] F. Hanzely et al. Fastest rates for stochastic mirror descent methods. Comput. Optim. Appl., 2021.

[7] L. Bottou, et al. Optimization methods for large-scale machine learning. SIAM Review, 2018.

[8] K. Wu, et al. Understanding stochastic natural gradient variational inference. ICML, 2024.

[9] C. Fang, et al. Sharp analysis for nonconvex sgd escaping from saddle points. PMLR, 2019.

[10] Boumal, N., et al. Deterministic Guarantees for Burer-Monteiro Factorizations of Smooth Semidefinite Programs. Comm. Pure Appl. Math., 2020.