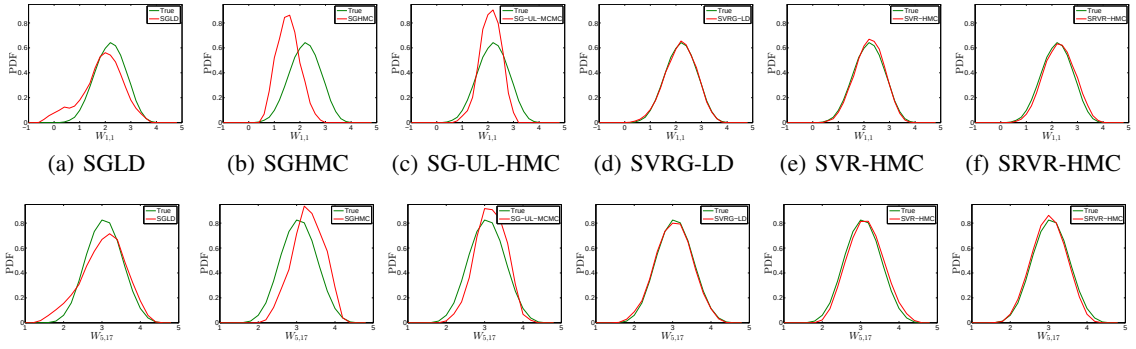
Response to Reviewer 1: Thank you for your supportive comments! We will fix the typos in the final version.

Response to Reviewer 2: Thanks for your helpful comments.

Q1: "The proposed algorithm is a relatively standard extension of SG-HMC and SGLD. While the proposed framework and analysis and bounds are a contribution, such results are fairly familiar in the literature."
A1: From the perspective of the design of our algorithm, we admit that our algorithm is an extension of SG-HMC. While SG-HMC and SGLD type algorithms have been widely studied in the literature, and might look familiar to you, our proposed algorithm is new, and its theoretical analysis has never been done in the literature and therefore is also new. More importantly, the corresponding theoretical guarantees of our algorithm outperform the state-of-the-art. This is also verified by our experiments.
Q2: "The following articles might also be related..."
A2: Thank you for pointing out these related articles. We will definitely cite and discuss them in the final version.
Q3: "Why not show figures that compare these samples against some ground truth, for example, those obtained by HMC (which is feasible to obtain for GMM and ICA)? See those of ...."
A3: Thank you for your suggestion to show the comparison between the samples generated by our method and the ground truth obtained by HMC. We have added this additional comparison in Figure 1 for ICA. In detail, we use HMC with metropolis hasting correction to generate the ground truth. Following the references pointed out by you, we randomly choose two variables ($W_{1,1}$ and $W_{5,17}$) from the parameter matrix $\mathbf{W}$ and display their marginal distributions after 1000 data passes in Figures 1(a)-1(f) (row 1) and Figures 1(g)-1(l) (row 2) respectively. It can be observed that the proposed SRVR-HMC (as well as SVRG-LD and SVR-HMC) can well approximate the ground truth, while SGLD and SGHMC cannot provide accurate approximation. This further validates the superior performance of SRVR-HMC and other variance reduced algorithms (SVRG-LD, SVR-HMC). We will add these experimental results in the final version.



(a) SGLD (b) SGHMC (c) SG-UL-HMC (d) SVRG-LD (e) SVR-HMC (f) SRVR-HMC

(g) SGLD (h) SGHMC (i) SG-UL-MCMC (j) SVRG-LD (k) SVR-HMC (l) SRVR-HMC

Figure 1: Marginal distributions of the posterior samples generated by Langevin dynamics based algorithms (red line) including SGLD, SGHMC, SG-UL-MCMC, SVRG-LD, SVR-HMC and SRVR-HMC, as well as the ground truth (green line).

Response to Reviewer 3: Thank you for your supportive comments.

Q1: "It's unclear if there is any way to do anything other than roughly estimate how that hyperparameter should be set."
A1: As suggested in Corollary 3.5 of our paper, the batch size parameter B should be smaller than $O(\epsilon^{-2}\mu_*^{-1/2} \wedge \sqrt{n})$, where ϵ is the precision parameter, μ_* is the spectral gap and n is the sample size of the dataset. For general non-log-concave distributions, the spectral gap μ_* can be very small, thus we often have $\epsilon^{-2}\mu_*^{-1/2} \geq \sqrt{n}$. Therefore, the batch size parameter in our experiments is chosen to guarantee that $B \leq C\sqrt{n}$ with C being a tuning parameter. This is validated by the sensitivity analysis of batch size B in Figures 2(b), 3(c) and 3(d) in our submission.
Q2: "The paper could have been improved a bit by maybe running the algorithm on slightly larger datasets."
A2: We indeed ran the algorithm on a large dataset (a9a, training sample size: 32561, test sample size: 16281) for Bayesian logistic regression. Due to the space limit, we put it in Appendix E. Follow your suggestion, we also ran additional experiments for ICA on a larger dataset (extract a larger subset from the original dataset, i.e., $n = 10000$), which is displayed in Figure 2. It can be seen that the proposed SRVR-HMC algorithm achieves the best performance among all methods. We will add more experiments on larger datasets in the appendix of the final version.

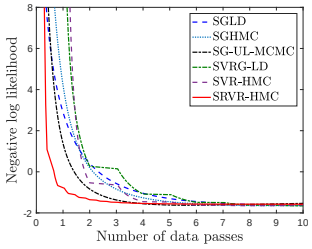


Figure 2: Results for ICA on a larger dataset (training sample size: $n = 10000$). X-axis represents the number of data passes and Y-axis represents the negative log likelihood on the test dataset.