Bayesian Learning via Stochastic Gradient Langevin Dynamics

Authors: Max Welling and Yee Whye Teh

Amal Chaoui and Narjisse Lasri
Master 2 Data Science

December 8, 2022

Outline

1 Introduction

2 Stochastic Gradient Langevin Dynamics

3 Contribution: Stochastic Gradient Hamiltonian Dynamics

4 Conclusion

Introduction

Large scale datasets limits.

- MCMC: each iteration requires computations over the **whole dataset**.
- SGD: does not capture parameter uncertainty and can potentially overfit data.
- Idea: **algorithm that overcomes MCMC and SGD issues.**

Optimization.

- The first phase of MCMC is optimization ("burn-in").
- Idea: algorithm that naturally transitions between optimization and sampling.

Noise generalizes well.

- Neural network.
- Idea: add noise into the algorithm for better generalization.

Stochastic Gradient Langevin Dynamics

SGLD: Theoretical setting

- **Key motivation.** Combine SGD and MCMC to benefit from the advantages of both methods.
- **Idea.** Algorithm with the proposed update :

$$\Delta\theta_t = \frac{\epsilon_t}{2}\left(\nabla \log p(\theta_t) + \frac{N}{n}\sum_{i=1}^{n}\nabla \log p(x_{ti} \mid \theta_t)\right) + \eta_t; \quad \eta_t \sim N(0, \epsilon_t I)$$

  - *Initial phase:* imitates an efficient SGD.
  - *Transition:* smooth transition from the burn-in optimisation phase into the posterior sampling phase.
  - *Second phase:* the injected noise dominates and the algorithm approximates MCMC.
- **Advantage.** Efficient use of large datasets while allowing for parameter uncertainty to be captured in a Bayesian manner.

SGLD: Application

- **Setting.** Bayesian logistic regression model with a Gaussian prior:

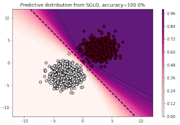
$$\theta \sim \mathcal{N}(0, I); \quad y_i \underset{iid}{\sim} \mathcal{B}(\sigma(y_i x_i^T \theta)), \quad \forall i \in \{1, \dots, N\}$$

.

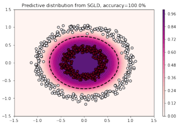
- **Gradient update.** $\epsilon_t = a(b+t)^{-\gamma}, \quad \eta_t \sim \mathcal{N}(O, \epsilon_t I)$

$$\theta_{t+1} = \theta_t + \frac{\epsilon_t}{2}\left[-\frac{1}{\sigma^2}\theta + \frac{N}{n}\sum_{i=1}^{n}(1-\sigma(y_i x_i^T \theta))y_i x_i\right] + \eta_t$$

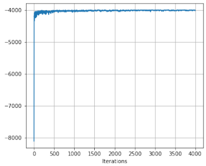
- Application to linear and circular **synthetic dataset**.
- Application to the **Bank Marketing Dataset**[1] (BMD).



(a) Linear



(b) Circular



(c) Log-posterior (BMD)

	SGLD	NUTS
Train acc.	**0.801 ± 0.001**	**0.801 ± 0.001**
Test acc.	0.788 ± 0.003	**0.790 ± 0.002**

(d) Accuracies (BMD)

[1] https://www.kaggle.com/datasets/janiobachmann/bank-marketing-dataset

Strengths. SGLD combines:

- Bayesian framework for uncertainty quantification and model regularisation.
- Gradient updates by mini-batches to reduce computation time.

Weaknesses.

- Need to tune the step size parameters a, b, γ to control the chaotic behaviour of convergence.
- **Poor mixing during the posterior sampling phase.**

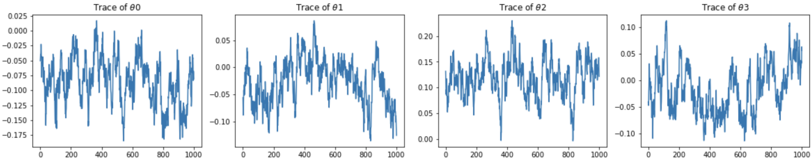


Figure 2: Trace plots of some parameters of θ obtained using SGLD.

Contribution: Stochastic Gradient Hamiltonian Dynamics

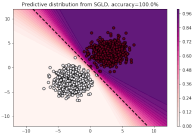
Contribution: Stochastic Gradient Hamiltonian Dynamics

- **Key motivation.** The Hamiltonian dynamic untangles the samples' correlation by introducing p: $\mathcal{H}(q,p) = -\log \pi(q) + p^T M_H p$.
- **Idea.** Use batch updates of the gradient in HMC.
  The leapfrog updates are computed on a mini-batch of n samples.

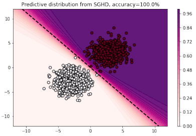
$$p_{1/2} = p + \frac{h}{2} \nabla_n \log \pi(q), \quad q' = q + hMp_{1/2}, \quad p' = p_{1/2} + \frac{h}{2} \nabla_n \log \pi(q').$$

- **Application.** Bayesian logistic regression.

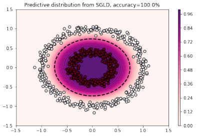
Params: $n_{batch1} = 750, n_{batch2} = 50, h = 0.5, n_h = 80.$



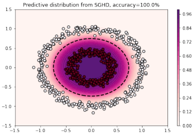
(a) SGLD



(b) SGHD



(c) SGLD



(d) SGHD

Contribution: Stochastic Gradient Hamiltonian Dynamics

Application to the Bank Marketing Dataset (BMD).

Params: $a = 0.01, b = 100, \gamma = 0.6, n_{batch} = 140, h = 0.5, n_h = 80.$

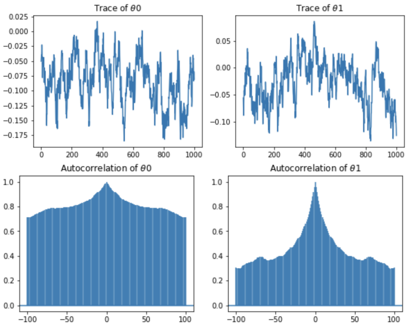


Figure 4: SGLD trace and auto-correlation plots.

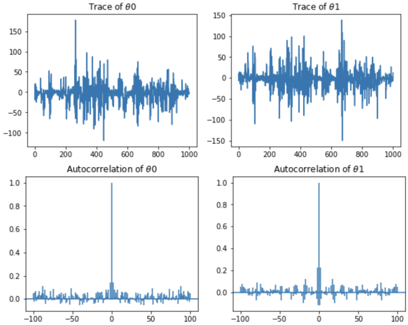


Figure 5: SGHD trace and auto-correlation plots.

	SGLD	**SGHD**	NUTS
Train acc.	0.801 ± 0.001	$\mathbf{0.713 \pm 0.042}$	0.801 ± 0.001
Test acc.	0.788 ± 0.003	$\mathbf{0.708 \pm 0.040}$	0.790 ± 0.002

Conclusion

Conclusion

SGLD.

- Combines stochastic optimization and sampling in one single algorithm.
- We obtain correlated chain moves.

SGHD.

- Uncorrelates the moves and improves the mixing while updating p, q on a mini-batch.
- Performance sensitive to the initial θ_0 and M_H.
- Acceptance probability needs to be computed over the whole data.

Perspectives.

- SGHD performance on multimodal posterior distributions.
- Integrate an optimization phase to SGHD.
- Parameters tuning: a, b, γ for SGLD, M_H, h, n_h for SGHD.

References

[1] Max Welling and Yee Whye Teh. "Bayesian Learning via Stochastic Gradient Langevin Dynamics". In: *Proceedings of the 28th International Conference on International Conference on Machine Learning*. ICML'11. Bellevue, Washington, USA: Omnipress, 2011, pp. 681–688. ISBN: 9781450306195.

[2] Radford Neal. "MCMC using Hamiltonian Dynamics". In: *Handbook of Markov Chain Monte Carlo* (June 2012). DOI: 10.1201/b10905-6.

Transition to the sampling phase: $\frac{\epsilon_t N^2}{4n}\lambda_{max}(V_s) = \alpha \ll 1$, with V_s being the empirical covariance of the scores $s_i = \nabla \log p(x_i|\theta_t) + \frac{1}{N}\nabla p(\theta_t)$.

Convergence w.r.t. step size: $b = 100, \gamma = 0.6$

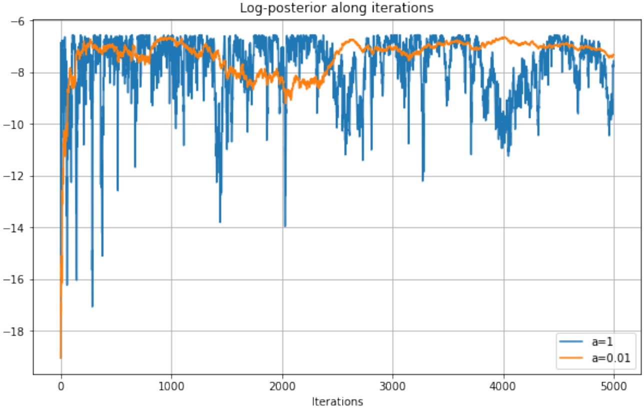
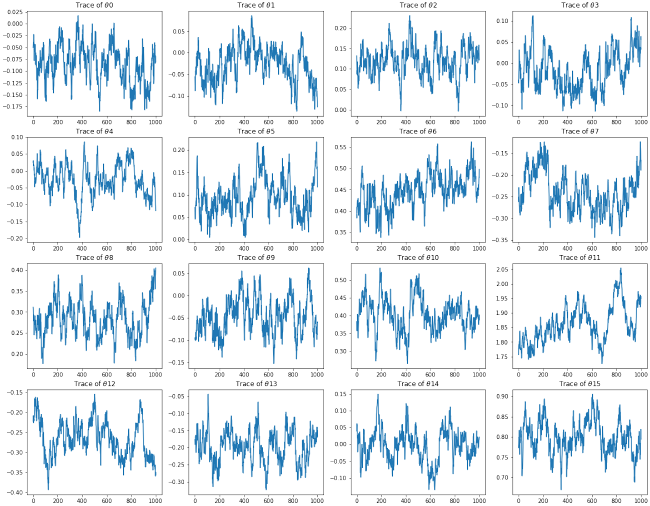


Figure 6: Log-posterior along iterations for small step sizes ($a = 0.01$) vs large step sizes ($a = 1$).

Appendix B

Trace plot of SGLD on the real dataset.



Appendix

Trace plot of SGLD on the real dataset.