

---

# Robust Approximate Sampling via Stochastic Gradient Barker Dynamics

---

**Lorenzo Mauri**

Department of Statistical Science  
Duke University

**Giacomo Zanella**

Department of Decision Sciences and BIDS  
Bocconi University

## Abstract

Stochastic Gradient (SG) Markov Chain Monte Carlo algorithms (MCMC) are popular algorithms for Bayesian sampling in the presence of large datasets. However, they come with little theoretical guarantees and assessing their empirical performances is non-trivial. In such context, it is crucial to develop algorithms that are robust to the choice of hyperparameters and to gradients heterogeneity since, in practice, both the choice of step-size and behaviour of target gradients induce hard-to-control biases in the invariant distribution. In this work we introduce the stochastic gradient Barker dynamics (SGBD) algorithm, extending the recently developed Barker MCMC scheme, a robust alternative to Langevin-based sampling algorithms, to the stochastic gradient framework. We characterize the impact of stochastic gradients on the Barker transition mechanism and develop a bias-corrected version that, under suitable assumptions, eliminates the error due to the gradient noise in the proposal. We illustrate the performance on a number of high-dimensional examples, showing that SGBD is more robust to hyperparameter tuning and to irregular behavior of the target gradients compared to the popular stochastic gradient Langevin dynamics algorithm.

## 1 INTRODUCTION

Approximating posterior distributions arising from probabilistic models is a challenging computational

---

Proceedings of the 27<sup>th</sup> International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238. Copyright 2024 by the author(s).

task, especially in the context of large datasets. Standard gradient-based MCMC algorithms (Roberts and Rosenthal, 1995; Duane et al., 1987; Neal, 2012) require evaluations of the exact target density and its gradient at each iteration, which can be computationally impractical. Inspired by stochastic optimization (Robbins and Monro, 1951), stochastic gradient MCMC (SG-MCMC) algorithms replace the exact target gradient with a computationally cheaper estimate, such as those obtained from a randomly sampled subset of the original data. Since the influential work of Welling and Teh (2011), which introduced the stochastic gradient Langevin dynamics (SGLD) algorithm, SG-MCMC methods have gained considerable popularity among practitioners seeking to perform approximate Bayesian inferences with large datasets. We refer to Nemeth and Fearnhead (2021) for an overview of SG-MCMC.

Most SG-MCMC methods Welling and Teh (2011); Chen et al. (2014); Ding et al. (2014); Ma et al. (2015) converge to the true posterior distribution if the step-size is appropriately decreased to zero (Teh et al., 2016). However, this strategy deteriorates mixing, increasing computational cost. Practitioners usually keep the step-size fixed, which leads to non-negligible and hard-to-diagnose bias in the invariant distribution (Betancourt, 2015; Brosse et al., 2018), especially if the step-size is chosen too large or the target distribution is irregular. Also, adaptive tuning is harder in the stochastic gradient setting relative to standard MCMC (Nemeth and Fearnhead, 2021), which makes robust methods even more appealing in this context. Motivated by these considerations, we develop the stochastic gradient version of the Barker proposal scheme developed in Livingstone and Zanella (2022), which has been shown to enjoy improved robustness to target heterogeneity and hyperparameter tuning relative to classical gradient-based MCMC schemes.

The paper is structured as follows. Section 2 sets up notation and provides background on the Barker proposal. Section 3 introduces and analyzes the stochas-

tic gradient Barker dynamics (SGBD) algorithm. In particular: Section 3.1 analyzes the bias induced by direct use stochastic gradients; Sections 3.2-3.3 propose a bias-correction methodology and identify the maximum level of noise that it can tolerate; Section 3.4 shows how to minimize bias for higher levels of noise. Section 4 numerically compares SGBD to SGLD. Therein, SGBD displays greater robustness to the choice of hyperparameters and to irregular posterior distributions, and in most cases exhibits either comparable or better out-of-sample predictive performance. Section 5 discusses future research directions.

## 2 BACKGROUND

We consider the task of approximate sampling from a target probability distribution of the form  $\pi(\theta) \propto \exp(g(\theta))$  where  $\theta \in \mathbb{R}^d$  and  $g: \mathbb{R}^d \rightarrow \mathbb{R}$ . In a classical Bayesian setting with conditional independent data, we have  $g(\theta) = \log(p(\theta)) + \sum_{i=1}^N \log(p(y_i | x_i, \theta))$ , where  $p(\theta)$  is the prior distribution of the parameter  $\theta$ , and  $p(y_i | x_i, \theta)$  is the likelihood component of the  $i$ -th data point  $y_i$  with covariates  $x_i$ . Hence, the gradient of  $g(\theta)$  can be written as the sum of  $N$  data points components,  $\partial_j g(\theta) = \sum_{i=1}^N \partial_j g_i(\theta)$ , where  $\partial_j g_i(\theta) = \frac{1}{N} \partial_j \log(p(\theta)) + \partial_j \log(p(y_i | x_i, \theta))$ , and  $\partial_j$  stands for the partial derivative with respect to the  $j^{\text{th}}$  component of  $\theta$ , i.e.  $\partial_j g(\cdot) = \frac{\partial}{\partial \theta_j} g(\cdot)$  for  $j = 1, \dots, d$ .

### 2.1 The Barker Proposal

The Barker proposal (Livingstone and Zanella, 2022; Hird et al., 2020) is a first-order approximation of a locally-balanced jump process (Zanella, 2020; Power and Goldman, 2019; Sun et al., 2023). The latter are continuous-time  $\pi$ -invariant jump processes with generator  $Lf(\theta) = \int (f(\theta+w) - f(\theta))J(\theta, \theta+w)dw$  defined by the intensity function

$$J(\theta, \theta + w) = h\left(\frac{\pi(\theta + w)}{\pi(\theta)}\right) \prod_{j=1}^d \mu_\sigma(w_j) \quad \theta, w \in \mathbb{R}^d.$$

Above  $h: (0, \infty) \rightarrow (0, \infty)$  can be any function satisfying  $h(t) = th(1/t)$  and  $\mu_\sigma(z) = \sigma^{-1}\mu(z/\sigma)$  any probability density function (PDF) on  $\mathbb{R}$  with scale parameter  $\sigma > 0$  and a symmetric reference distribution  $\mu$ . Taking  $h$  as the Barker function,  $h(t) = 2t(1+t)^{-1}$ , and approximating  $\pi$  with its first order log-Taylor expansion,  $\pi(\theta+w)/\pi(\theta) \approx \exp(\sum_{j=1}^d \partial_j g(\theta)w_j)$  leads to

the Barker proposal, whose PDF is

$$Q_B(\theta, \theta + w) = \prod_{i=1}^d 2p(\partial_j g(\theta), w_j) \mu_\sigma(w_j) \quad \theta, w \in \mathbb{R}^d \quad (1)$$

$$p(\delta, z) = (1 + \exp(-z\delta))^{-1} \quad \delta, z \in \mathbb{R}. \quad (2)$$

Since  $p(\delta, z) + p(\delta, -z) = 1$ ,  $Q_B$  is a product of skew-symmetric distributions (Azzalini, 2013), which provides full-tractability as well as a straightforward algorithm to sample from  $Q_B$  (namely lines 2-7 in Algorithm 1). The gradient  $\partial_j g(\theta^{(t-1)})$  enters into  $Q_B$  as a degree of skewness, as opposed to a linear shift of the mean as for classical Euler-Maruyama based schemes such as the Unadjusted Langevin Algorithm (ULA; see e.g. Roberts and Tweedie, 1996). It follows that the gradients only influence the direction of the increment under  $Q_B$  and not its size, since the distribution of  $|w_j|$  is independent of  $\partial_j g(\theta)$ . This decoupling of gradients and increments size leads to an increased robustness to sub-optimal hyperparameter tuning and target heterogeneity, see e.g. Livingstone and Zanella (2022) for more details and some formal results. On the other hand, being a first-order approximation to a  $\pi$ -invariant process,  $Q_B$  shares the favourable high-dimensional scaling properties of classical gradient-based MCMC, such as a scaling of order  $d^{-1/3}$  as  $d$  diverges (Roberts and Rosenthal, 1995; Voigrinc et al., 2022).

Algorithm 1 describes the resulting unadjusted Barker proposal algorithm. The key step is the flipping

---

#### Algorithm 1: Unadjusted Barker Proposal

---

**Input:**  $\theta^{(0)} \in \mathbb{R}^d, \sigma > 0$

```

1 for  $t=1, \dots, T$  do
2   for  $j=1, \dots, d$ ; // Can be parallelized
3   do
4     Draw  $w_j^{(t)} \sim \mu_\sigma(\cdot)$ ;
5     Set  $b_j^{(t)} = 1$  with probability
        $p(\partial_j g(\theta^{(t-1)}), w_j^{(t)})$ , otherwise  $b_j^{(t)} = -1$ ;
6     Update  $\theta_j^{(t)} \leftarrow \theta_j^{(t-1)} + b_j^{(t)} w_j^{(t)}$ ;
7   end
8 end
```

---

operation at line 5, where the algorithm flips the sign of the proposed increment with probability  $1 - p(\partial_j g(\theta), w_j^{(t)})$ . This operation skews the proposal distribution towards the target  $\pi$ , since the increment will be  $+w_j^{(t)}$  with high probability if  $\partial_j g(\theta)w_j^{(t)}$  is high and  $-w_j^{(t)}$  otherwise. Note that Algorithm 1 has essentially the same computational cost of standard

ULA, where  $\theta_j^{(t)} \leftarrow \theta_j^{(t-1)} + \sigma^2/2\partial_j g(\theta^{(t-1)}) + w_j^{(t)}$  and  $w_j^{(t)} \sim N(0, \sigma^2)$  for  $j = 1, \dots, d$ . Both schemes require  $O(N)$  operations at each iteration, with the computation of the gradient representing the major computational bottleneck with large datasets, as in other gradient-based schemes. Previous work have considered the Metropolis-adjusted version of Algorithm 1. Here we consider the unadjusted one, in order to maintain the computational savings induced by (3) when moving to the stochastic gradient context. While there are works combining MH schemes with mini-batching (Korattikara et al., 2014; Bardenet et al., 2014), stochastic-gradient versions of unadjusted schemes are much more common and widely used.

In our experiments below we take  $\mu_\sigma$  to be the bimodal distribution  $0.5N(-\sigma, (0.1\sigma)^2) + 0.5N(\sigma, (0.1\sigma)^2)$ , as recommended in (Vogrinic et al., 2022). Note that in algorithmic implementations one can simply take  $\mu_\sigma = N(\sigma, (0.1\sigma)^2)$ , since the resulting algorithm is equivalent by symmetry (though for the equality (1) to be correct one needs the symmetric version of  $\mu_\sigma$ ).

### 3 THE STOCHASTIC GRADIENT BARKER PROPOSAL (SGBD)

In this section we propose and analyze the stochastic-gradient Barker Proposal (SGBD) algorithm. At each iteration, we replace the true gradient with the mini-batch estimate

$$\hat{\partial}_j g(\theta) = \frac{N}{n} \sum_{i \in \mathcal{S}_n} \partial_j g_i(\theta) \quad j = 1, \dots, p, \quad (3)$$

where  $\mathcal{S}_n$  is a subset of  $\{1, \dots, N\}$  of size  $n \ll N$  sampled uniformly at random, with or without replacement. The vanilla version of SGBD (v-SGBD) consists in substituting the gradient in Algorithm 1 with the estimate in (3), leading to Algorithm 2.

#### 3.1 Bias of vanilla SGBD

Lines 2, 5 and 6 of Algorithm 2 are equivalent to setting  $b_j^{(t)} = 1$  with probability  $\mathbb{E}[p(\hat{\partial}_j g(\theta^{(t-1)}), w_j^{(t)})]$ , where the expectation is taken with respect to the subsampling mechanism. Thus, denoting current location and proposed increment as  $\theta \in \mathbb{R}^d$  and  $z \in \mathbb{R}$  for notational simplicity, Algorithm 2 effectively replaces the flipping probability, i.e.  $\Pr(b = 1) = p(\partial_j g(\theta), z)$  in Algorithm 1, with  $\mathbb{E}[p(\hat{\partial}_j g(\theta), z)]$ . While (3) ensures that  $\mathbb{E}[\hat{\partial}_j g(\theta)] = \partial_j g(\theta)$ , the non-linearity of  $p$  implies that  $\mathbb{E}[p(\hat{\partial}_j g(\theta), z)] \neq p(\partial_j g(\theta), z)$  in general. The bias of  $p(\hat{\partial}_j g(\theta), z)$  implies that the gradient noise does not balance out and, similarly to other SG-MCMC methods, Algorithm 2 introduces additional error in the

---

#### Algorithm 2: Vanilla Stochastic Gradient Barker Dynamics (v-SGBD)

---

**Input:**  $\theta^{(0)} \in \mathbb{R}^d, \sigma > 0$

- 1 **for**  $t=1, \dots, T$  **do**
- 2     Draw  $\mathcal{S}_n \subset \{1, \dots, N\}$  uniformly at random
- 3     **for**  $j=1, \dots, d$ ; // Can be parallelized
- 4     **do**
- 5         Draw  $w_j^{(t)} \sim N(\sigma, (0.1\sigma)^2)$ ;
- 6         Compute  $\hat{\partial}_j g(\theta^{(t-1)})$  as in (3);
- 7         Set  $b_j^{(t)} = 1$  with probability
- 8          $p(\hat{\partial}_j g(\theta^{(t-1)}), w_j^{(t)})$ , otherwise  $b_j^{(t)} = -1$ ;
- 9         Update  $\theta_j^{(t)} \leftarrow \theta_j^{(t-1)} + b_j^{(t)} w_j^{(t)}$ ;
- 10     **end**

---

stationary distribution of Algorithm 1. In the next section we analyse this bias and develop a strategy to reduce it. We implicitly assume that the approximation error inherent to Algorithm 1, induced by the first-order approximation of the jump process, is of smaller order relative to the one induced by stochastic gradients. The same holds true for most SG-MCMC schemes (Teh et al., 2016; Brosse et al., 2018).

First, we identify the direction of the bias. We make the following symmetry assumption on the stochastic gradient noise, which we denote as  $\eta_\theta := \hat{\partial}_j g(\theta) - \partial_j g(\theta)$ , suppressing the dependence on  $j$  for brevity.

**Condition 1** (Symmetry of  $\eta_\theta$ ).

$$\eta_\theta \stackrel{d}{=} -\eta_\theta, \quad (4)$$

where  $\stackrel{d}{=}$  denotes equality in distribution.

**Proposition 1** (Direction of bias). *Under Condition 1 we have*

$$|p(\partial_j g(\theta), z) - 0.5| \geq \left| \mathbb{E} \left[ p(\hat{\partial}_j g(\theta), z) \right] - 0.5 \right|. \quad (5)$$

Proposition 1, as well as results below, holds for every  $\theta \in \mathbb{R}^d$  and  $z \in \mathbb{R}$ . Proofs of all theoretical results are provided in the supplement. Proposition 1 shows that, under symmetric noise, the expectation of  $p(\hat{\partial}_j g(\theta), z)$  is always shrunk towards 0.5 relative to its target value  $p(\partial_j g(\theta), z)$ . The practical implication of (5) is an inflation of the variance of the stationary distribution, as Algorithm 2 moves less frequently towards a local mode of the distribution relative to Algorithm 1. This is analogous to what happens with other SG-MCMC algorithms: for instance, when the step-size is held fixed, the stochastic gradient noise increases the variance of the invariant distribution of SGLD when no correction is taken into account (Vollmer et al., 2016).

### 3.2 Corrected SGBD

In this section we quantify the bias of  $p(\hat{\partial}_j g(\theta), z)$  and derive a corrected estimator for  $\Pr(b = 1) = p(\partial_j g(\theta), z)$ . To do so, we assume the stochastic gradient noise to be normally distributed. This is a common requirement in the SG-MCMC theory literature, which is typically justified by assuming the mini-batch size to be sufficiently large and applying a central limit theorem (Chen et al., 2014; Ding et al., 2014; Lu et al., 2017).

**Condition 2** (Normality of  $\eta_\theta$ ).

$$\eta_\theta \sim \mathcal{N}(0, \tau_\theta^2), \quad (6)$$

for some  $\tau_\theta > 0$  that can depend on  $\theta$  (and on  $j$ ).

Under Condition 2, we obtain the following tractable approximation to the expectation of  $p(\hat{\partial}_j g(\theta), z)$ .

**Proposition 2.** Under Condition 2, we have

$$\left| \mathbb{E} \left[ p \left( \hat{\partial}_j g(\theta), z \right) \right] - p(c_{z, \tau_\theta} \partial_j g(\theta), z) \right| < 0.019, \quad (7)$$

$$\text{where } c_{z, \tau_\theta} := \frac{1.702}{\sqrt{1.702^2 + z^2 \tau_\theta^2}}.$$

**Remark 1.** Proposition 2 approximates Algorithm 2 by Algorithm 1 with target gradients shrunk by the multiplicative factor  $c_{z, \tau_\theta} < 1$ . This supports the idea that stochastic gradients have the effect of tempering the stationary distribution by a power smaller than 1, and suggests the strategy of multiplying them by a factor larger than 1 to counterbalance the effect induced by the noise. In particular, multiplying  $\hat{\partial}_j g(\theta)$  by  $\alpha > 1$  inflates its expectation by  $\alpha$  and its variance by  $\alpha^2$ . It turns out that the value  $\alpha = 1.702(1.702^2 - \tau_\theta^2 z^2)^{-1/2}$  (despite not depending on  $\partial_j g(\theta)$ ) makes the expectation of the resulting (corrected) estimator approximately equal to  $p$  with the correct partial derivative  $\partial_j g(\theta)$ . This is formalized in Corollary 1.

Following Remark 1, we define the corrected estimator of  $p(\partial_j g(\theta), z)$  as  $\tilde{p}(\hat{\partial}_j g(\theta), z)$ , where for any  $\delta, z \in \mathbb{R}$

$$\tilde{p}(\delta, z) := \begin{cases} p \left( \frac{1.702}{\sqrt{1.702^2 - \tau_\theta^2 z^2}} \delta, z \right) & \text{if } |z| < \frac{1.702}{\tau_\theta}, \\ \mathbf{1}(\delta z > 0) & \text{otherwise} \end{cases}, \quad (8)$$

with  $\mathbf{1}(A)$  denoting the indicator function of the event  $A$ . When the value of  $\tau_\theta$  is not too large,  $\tilde{p}(\hat{\partial}_j g(\theta), z)$  is an approximately unbiased estimator of  $p(\partial_j g(\theta), z)$ , as stated in the following corollary.

**Corollary 1** (Approximate unbiasedness of  $\tilde{p}$ ). Assume Condition 2 and  $\tau_\theta < \max\{1.702/|z|, \bar{\tau}(\partial_j g(\theta), z)\}$ , where

$$\bar{\tau}(\delta, z) = \left| \delta / \Phi^{-1} \left( (1 + \exp(-z\delta))^{-1} \right) \right| \quad (9)$$

and  $\Phi$  denotes the standard Normal CDF. Then

$$\left| \mathbb{E} \left[ \tilde{p} \left( \hat{\partial}_j g(\theta), z \right) \right] - p(\partial_j g(\theta), z) \right| < 0.019. \quad (10)$$

Replacing the naive estimate  $p(\hat{\partial}_j g(\theta), z)$  used in v-SGBD with  $\tilde{p}(\hat{\partial}_j g(\theta), z)$  leads to what we refer to as *corrected SGBD* (c-SGBD). Note, however, that the corrected estimator requires knowledge of the variance of the gradient noise,  $\tau_\theta$ . In practical applications,  $\tau_\theta$  must be estimated. To do that we adopt a simple on-line sample variance estimator, leading to the version of c-SGBD described in Algorithm 3.

---

**Algorithm 3:** Corrected Stochastic Gradient Barker Dynamics (c-SGBD)

---

**input:**  $\theta^{(0)} \in \mathbb{R}^d, \sigma > 0, \beta \in (0, 1), \{\hat{\tau}_j^{(0)}\}_{j=1, \dots, d}$

```

1 for  $t=1, \dots, T$  do
2   Draw  $\mathcal{S}_n \subset \{1, \dots, N\}$  uniformly at random;
3   for  $j=1, \dots, d$ ; // Can be parallelized
4   do
5     Compute  $\hat{\partial}_j g(\theta^{(t-1)})$  using (3);
6     Update  $\hat{\tau}_j^{(t)} \leftarrow (1 - \beta) \hat{\tau}_j^{(t-1)} +$ 
        $\beta \sqrt{\sum_{i \in \mathcal{S}_n} \frac{(\partial_j g_i(\theta^{(t-1)}) - \frac{1}{n} \sum_{i \in \mathcal{S}_n} \partial_j g_i(\theta^{(t-1)}))^2}{n-1}}$ ;
7     Draw  $w_j^{(t)} \sim N(\sigma, (0.1\sigma)^2)$ ;
8     Set  $b_j^{(t)} = 1$  with probability
        $\tilde{p}(\hat{\partial}_j g(\theta^{(t-1)}), w_j^{(t)})$ , where  $\tau_\theta$  in (8) is
       replaced by  $\hat{\tau}_j^{(t)}$ , otherwise  $b_j^{(t)} = -1$ ;
9     Update  $\theta_j^{(t)} \leftarrow \theta_j^{(t-1)} + b_j^{(t)} w_j^{(t)}$ ;
10  end
11 end

```

---

**Remark 2.** While Condition 2 is not satisfied in most practical scenarios, it allows to quantify the bias and devise estimators that can reduce it when normality holds approximately. See for example Figure 1.

**Remark 3.** The results in Proposition 2 and Corollary 1 are based on the bound  $\max_x |F(x) - \Phi(x/1.702)| < 0.0095$ , see e.g. Bowling et al. (2009, Section 3.2). Here  $F(\cdot)$  is the CDF of the logistic distribution and  $\Phi(\cdot)$  is the one of the standard Normal distribution.

### 3.3 Noise tolerance of SGBD

There is a maximum amount of noise that can be tolerated while still being able to estimate  $p(\partial_j g(\theta), z)$  from  $\hat{\partial}_j g(\theta)$ . In particular, even under Condition 2, if  $\tau_\theta$  is too large it is not possible to have unbiased estimators of  $p(\partial_j g(\theta), z)$ , by which we mean functions

$\hat{p}(\hat{\delta}, z; \tau_\theta)$  taking values in  $[0, 1]$  such that

$$\mathbb{E} \left[ \hat{p}(\hat{\delta}, z; \tau_\theta) \right] = p(\delta, z) \quad \text{for all } \delta \in \mathbb{R} \quad (11)$$

with expectation taken under  $\hat{\delta} \sim N(\delta, \tau_\theta^2)$ .

**Proposition 3** (Noise tolerance). *Assume Condition 2 and  $\tau_\theta > \tau^*$ , where*

$$\tau^* = \inf_{\delta \in \mathbb{R}} \bar{\tau}(\delta, z) = 4\phi(0)/|z|,$$

with  $\phi$  denoting the standard Normal PDF. Then there exist no unbiased estimator of  $p(\partial_j g(\theta), z)$ .

Note that  $[0, 1]$ -valued unbiased estimators are what is needed in order to implement a stochastic gradient scheme that introduce no further bias in Algorithm 1. Thus, Proposition 3 identifies a noise level,  $\tau^*$ , beyond which it is not possible to implement SGBD without introducing further bias due to stochastic gradients. The value of  $\tau^* = 4\phi(0)/|z| \approx 1.596/|z|$  is related to, though slightly smaller than, the upper bound on  $\tau_\theta$  we required in Corollary 1 to guarantee approximate unbiasedness of  $\tilde{p}(\hat{\partial}_j g(\theta), z)$ . Intuitively, since only approximate unbiasedness is required in Corollary 1, one can afford slightly larger values of  $\tau_\theta$  therein.

We can re-interpret Proposition 3 in terms of upper bound to the algorithmic step-size: given a noise level  $\tau_\theta$ , the largest increment one can propose in Algorithm 3 without introducing further bias due to stochastic gradients is  $|z| \leq 1.596/\tau_\theta$  (or  $|z| \leq 1.702/\tau_\theta$  if a small controllable bias is allowed as in Corollary 1). These results could also be used to devise adaptive versions of SGBD where  $\sigma$  is tuned on-the-fly so that  $|z| \leq 1.702/\tau_\theta$  occurs with high-probability, in a spirit similar to e.g. Ding et al. (2014). We leave such extensions to future work.

Figure 1 numerically illustrates these phenomena. There we plot  $p(\partial_j g(\theta), z)$ ,  $\mathbb{E}[p(\hat{\partial}_j g(\theta), z)]$  and  $\mathbb{E}[\tilde{p}(\hat{\partial}_j g(\theta), z)]$  as a function of  $z$  when  $\pi(\theta)$  is the posterior distribution in a high-dimensional logistic regression model with real data and randomly chosen values of  $\theta$  and  $j$ . See the supplement for more details on the model and generation of  $\theta$  and  $j$ . We observe the value of  $\mathbb{E}[p(\hat{\partial}_j g(\theta), z)]$  being close to  $p(\partial_j g(\theta), z)$  when  $|z|$  is small, while as  $|z|$  increases the shrinkage effect discussed in Proposition 1 becomes evident. The corrected estimator  $\tilde{p}$  successfully reduces the bias up until the tolerance level  $|z| \approx 1.702/\tau_\theta$ , after which the signal in the stochastic gradient is too weak to successfully estimate the true value of  $p$  (Corollary 1 and Proposition 3). We refer to the supplement for a comparison between the tolerance level of SGBD and SGLD.

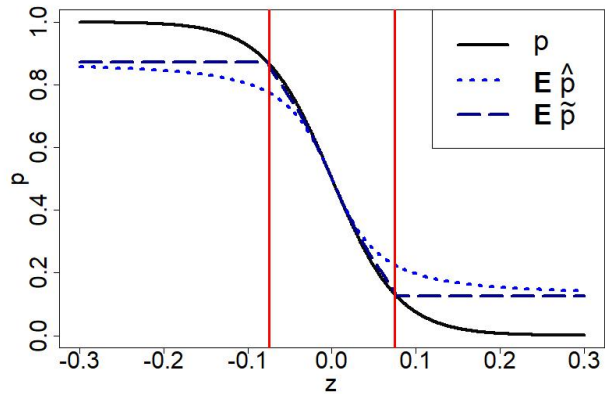


Figure 1. Shrinkage effect and bias correction. Plot of  $p(\partial_j g(\theta), z)$  (black line;  $p$ ) and Monte Carlo estimates of  $\mathbb{E}[p(\hat{\partial}_j g(\theta), z)]$  (dotted blue line;  $\mathbf{E}\hat{p}$ ), and  $\mathbb{E}[\tilde{p}(\hat{\partial}_j g(\theta), z)]$  (dashed dark blue line;  $\mathbf{E}\tilde{p}$ ) versus the proposed increment  $z$ ; for a logistic regression example with real data (see supplement for more details). Vertical red lines indicate  $-1.702/\tau_\theta$  and  $1.702/\tau_\theta$ .

### 3.4 Extreme SGBD

Sections 3.2 and 3.3 show that, under Condition 2, one can implement a stochastic-gradient version of the unadjusted Barker proposal without introducing significant bias. Doing that requires  $\tau_\theta < \max\{1.702/|z|, \bar{\tau}(\partial_j g(\theta), z)\}$ . This can be achieved either by reducing the stepsize  $\sigma$  (which reduces  $|z|$ ) or by increasing the minibatch size  $n$  (which reduces  $\tau_\theta$ ). However, in many settings, users prefer to run SGMCMC schemes with larger stepsize and smaller minibatch size to speed-up convergence and reduce computational cost, even if this introduces non-negligible bias. In this section we thus focus on the case of larger values of  $\tau_\theta$  and identify the optimal estimator of  $p(\partial_j g(\theta), z)$  in such settings, which turns out to be

$$\bar{p}(\delta, z) := \mathbf{1}(\delta z > 0),$$

i.e.  $\bar{p}(\delta, z)$  equals 1 when  $\delta$  and  $z$  have the same sign and 0 otherwise. We refer to  $\bar{p}$  as *extreme* estimator. Note that  $\bar{p}$  coincides with the *corrected*-estimator,  $\tilde{p}$ , whenever  $\tau_\theta \geq 1.702/|z|$ . We will prove optimality of  $\bar{p}$  within the following class of estimators.

**Definition 1** (Symmetric estimator). *An estimator  $\hat{p}(\hat{\delta}, z) = \hat{p}(\hat{\delta}, z; \tau_\theta)$  of  $p(\partial_j g(\theta), z)$  is said to be symmetric if*

$$\hat{p}(\hat{\delta}, z) + \hat{p}(-\hat{\delta}, z) = 1 \quad \text{for all } (\hat{\delta}, z) \in \mathbb{R}^2 \quad (12)$$

Condition (12) requires  $\hat{p}$  to have a symmetric behaviour about 0.5 when the sign of the stochastic gra-

dient is flipped, i.e.  $\hat{p}(\hat{\delta}, z) - 0.5 = 0.5 - \hat{p}(-\hat{\delta}, z)$ . The latter is a natural requirement for an estimator of  $p(\partial_j g(\theta), z)$  to be of practical interest, otherwise the resulting algorithm would be unjustifiably biased towards the left or right side of the current value of  $\theta_j$ .

We make the following assumption on stochastic gradients, which is strictly weaker than Condition 2.

**Condition 3** (Unimodality of noise distribution). *The random variable  $\eta_\theta$  admits a density function  $f_\theta(x)$  with respect to the Lebesgue measure and  $f_\theta(x)$  is non-decreasing for  $x \leq 0$  and non-increasing otherwise.*

Relative to Condition 2, Conditions 1 and 3 accommodate for more general scenarios, such as heavier-tailed distributions of  $\eta_\theta$ . Under these conditions, any symmetric estimator induces more shrinkage towards 0.5 than  $\bar{p}$  as we now show.

**Proposition 4.** *Under Conditions 1 and 3, we have*

$$\left| \mathbb{E} \left[ \bar{p} \left( \hat{\partial}_j g(\theta), z \right) \right] - 0.5 \right| \geq \left| \mathbb{E} \left[ \hat{p} \left( \hat{\partial}_j g(\theta), z \right) \right] - 0.5 \right|. \quad (13)$$

for any symmetric estimator  $\hat{p}$ , with strict inequality when  $\hat{p} \neq \bar{p}$  and  $\partial_j g(\theta)z \neq 0$ .

Since  $\hat{p}$  is always biased towards 0.5 for large  $\tau_\theta$ , Proposition 4 implies that in such cases  $\bar{p}$  achieves minimal bias. We formally state this under Condition 2, quantifying how large  $\tau_\theta$  needs to be.

**Corollary 2.** *Assume Condition 2,  $\partial_j g(\theta)z \neq 0$ , and  $\tau_\theta > \bar{\tau}(\partial_j g(\theta), z)$ , with  $\bar{\tau}$  defined in (9). Then*

$$\begin{aligned}
 & \left| p(\partial_j g(\theta), z) - \mathbb{E} \left[ \bar{p} \left( \hat{\partial}_j g(\theta), z \right) \right] \right| \\
 & < \left| p(\partial_j g(\theta), z) - \mathbb{E} \left[ \hat{p} \left( \hat{\partial}_j g(\theta), z \right) \right] \right|
 \end{aligned}$$

for any symmetric estimator  $\hat{p} \neq \bar{p}$ .

Corollary 2 supports the use of  $\bar{p}$  for large values of  $\tau_\theta$ . Also, Corollaries 1 and 2 combined imply that the *corrected* estimator  $\hat{p}(\hat{\partial}_j g(\theta), z)$  is optimal, in the sense of being approximately unbiased when  $\tau_\theta \leq \bar{\tau}(\partial_j g(\theta), z)$  and achieving minimal bias when  $\tau_\theta > \bar{\tau}(\partial_j g(\theta), z)$ . In our simulations, especially when we are interested in predictive accuracy, we also study the performance of the algorithm that always employs the extreme estimator, irrespective of the value of  $z$  and  $\tau_\theta$ . We refer to such algorithm as *extreme* SGBD (e-SGBD), and provide pseudo-code for it in the supplement. Effectively, e-SGBD sets  $b = 1$  when  $z$  and  $\hat{\partial}_j g(\theta^{(t-1)})$  have the same sign and  $b = -1$  otherwise. Hence, it always moves each coordinate in the direction of its component of the stochastic gradient, in a way that is similar to stochastic optimization methods such as AdaGrad (Duchi et al., 2011).

## 4 EXPERIMENTS

In this section we study the performances of the three SGBD versions (vanilla, corrected and extreme), applying them to sampling tasks arising from various models and comparing them to SGLD. To help comparability, we consider three versions of SGLD: a vanilla one (v-SGLD) corresponding to the stochastic gradient version of ULA; a corrected one (c-SGLD; see Algorithm 4 in the supplement), where the standard deviation of the artificial noise is adjusted to correct for the stochastic gradient noise; and an extreme one (e-SGLD), where the maximum correction is applied by adding no artificial noise (which simply corresponds to the stochastic gradient descent algorithm, Robbins and Monro, 1951). We also tested the modified variant of SGLD proposed in (Vollmer et al., 2016), obtaining comparable results to c-SGLD. Full details on the SGLD variants, as well as additional simulations for this section can be found in the supplement. Overall, the results support the fact that the increased robustness of the Barker scheme to target heterogeneity and hyperparameters tuning (Livingstone and Zanella, 2022) is relevant also in the stochastic gradient context.

### 4.1 Skewed target distributions

First, we study how skewness in the target distribution affects algorithmic performances. Skewness naturally generates heterogeneity in the magnitude of the gradient  $\partial_j g(\theta)$  in different regions of the state space, thus being an interesting test case for SG-MCMC schemes. In particular, we consider the family of skew-normal target distributions, i.e.  $\pi_\alpha(\theta) = 2\phi(\theta)\Phi(\alpha\theta)$  where  $\theta \in \mathbb{R}$ ,  $\alpha > 0$  and  $\phi$  and  $\Phi$  are the standard normal PDF and CDF, for different values of the skewness parameter  $\alpha$ . We take  $\eta_\theta \sim N(0, \text{var}(\pi_\alpha))$  and consider two values for the step-size  $\sigma$ , namely  $\sigma_1 = 0.1 \times \text{sd}(\pi_\alpha)$  and  $\sigma_2 = 0.5 \times \text{sd}(\pi_\alpha)$ , where  $\text{sd}(\pi_\alpha)$  and  $\text{var}(\pi_\alpha)$  denote the standard deviation and variance of  $\pi_\alpha$ . Figure 2 displays the results for v-SGBD and v-SGLD. The results obtained with corrected variants, as well as different noise levels for  $\eta_\theta$ , led to similar conclusions and are reported in the supplement. Figure 2a reports the bias on the posterior mean estimates as a function of the skewness parameter  $\alpha$ , while Figure 2b displays stationary distribution for two values of  $\alpha$ . The results suggest that SGBD is more robust to skewness relative to SGLD, which can suffer from high bias in the invariant distribution as  $\alpha$  increases. Also, the two algorithms exhibit a different robustness to hyperparameter tuning, with SGBD displaying more stable performances when the step-size is increased.

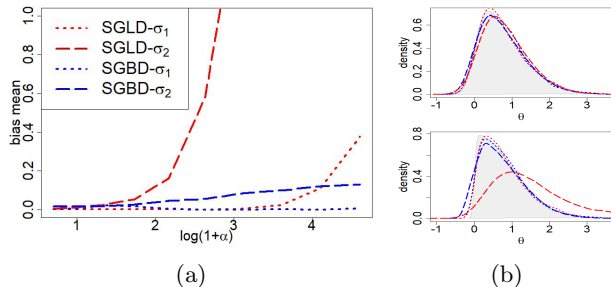


Figure 2. Toy Example: Skew-Normal target with isotropic Gaussian Noise. Shape parameter (in log-scale) vs relative bias of mean (left) and invariant distribution of the samplers for different levels of  $\alpha$ : 5 (top) and 20 (bottom) (right). Red refers to vanilla Langevin-based schemes and blue to vanilla Barker-based schemes. Dotted (dashed resp.) lines are produced with  $\sigma_1 = 0.1 \times sd(\pi_\alpha)$  ( $\sigma_2 = 0.5 \times sd(\pi_\alpha)$  resp.) of the target standard deviation. The grey shaded area in the right plots represents the true target distribution density.

## 4.2 Scale Heterogeneity

Next, we study the performance of SGBD on a binary regression task with scale heterogeneity across parameters. We consider a Bayesian logistic regression model of the form

$$y_i | x_i, \theta \sim \text{Bern}\left(\left(1 + e^{(-\theta^\top x_i)}\right)^{-1}\right) \quad i = 1, \dots, N, \quad (14)$$

where  $x_i \in \mathbb{R}^d$  and  $y_i \in \{0, 1\}$ . We use SG-MCMC to target the posterior distribution  $\pi(\theta) = p(\theta | x, y)$  resulting from (14) and a standard normal prior on  $\theta$ , i.e.  $\theta \sim \mathcal{N}_d(0, \mathbf{I}_d)$ . We consider the Sepsis dataset from the UCI repository, which contains  $N = 110204$  instances and  $d = 4$  covariates. Note that this example is low-dimensional, while in the supplement a high-dimensional binary regression data-set is considered. The Sepsis dataset leads to an ill-conditioned posterior distribution, where the posterior standard deviation of the first coordinate is much smaller than the others, as shown in Figure 3. While similar issues related to ill-conditioning could be in principle solved by preconditioning, finding good posterior preconditioners before running MCMC is not always easy and in practice SG-MCMC schemes are often used without adaptive preconditioning, also due to the difficulty in tuning them (Nemeth and Fearnhead, 2021). In this sense, robustness to scale heterogeneity is a desirable feature for SG-MCMC schemes.

We compare v-SGBD and v-SGLD, with a mini-batch size of  $n = \lfloor 0.01N \rfloor$  and  $T = 2 \times 10^5$  iterations. Figure 3 reports the resulting traceplots in the stationary phase. Black lines correspond to the true posterior mean plus and minus two standard deviations, com-

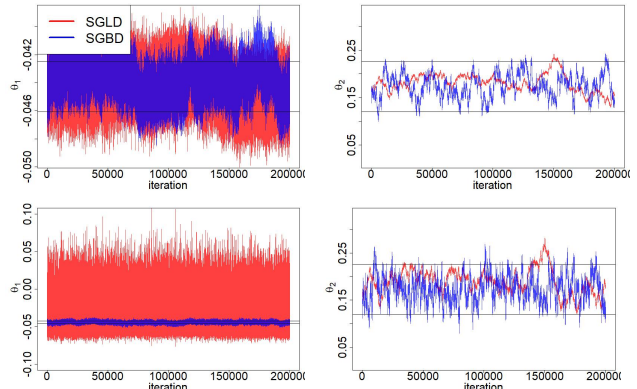


Figure 3. Sepsis dataset example. Traceplots of two coordinates with a different scale ( $\theta_1$  on the left has small scale) with two step-size configuration: small  $\sigma$  (top), larger  $\sigma$  (bottom). Red refers to v-SGLD and blue to v-SGBD. Black horizontal lines indicate the interval centered at the posterior mean with two standard deviations width.

puted with full-batch MCMC using Stan (Carpenter et al., 2017).

Due to ill-conditioning, SGLD struggles to explore the posterior: tuning the step-size to match the first coordinate (Figure 3 top row) leads to very poor mixing in the other coordinates; while doubling the step-size with respect to that (Figure 3 bottom row) dramatically inflates the variance of the first component and leads to biased inferences. Instead SGBD, while still being affected by ill-conditioning, is remarkably more robust to scale heterogeneity (e.g. it simultaneously achieves good mixing and moderate bias in the stationary distribution) as well as to the choice of hyperparameters (e.g. doubling the step-size from top to bottom only leads to a moderate inflation of the marginal stationary distribution in the first component). The specific values for the hyperparameters used in Figure 3 are  $\sigma = 0.00075, 0.0015$  for SGBD and  $\sigma = 0.0002, 0.0004$  for SGLD. See also the supplement for plots of marginal density estimates.

In the supplement we fit model (14) to the Arrhythmia data-set which contains 100 features and 362 training data points. In this example, SGLD is more accurate for small step-sizes and low mixing, while SGBD appears to be more robust to hyperparameter tuning. In particular, it enjoys a more favourable mixing-accuracy trade off when the step size is chosen increasingly large. In addition, SGBD obtains a better prediction accuracy on the hold-out set for all hyperparameter configurations considered.

### 4.3 Bayesian Matrix Factorization

We consider a Bayesian matrix factorization model (Salakhutdinov and Mnih, 2008) for recommendation. We have  $U$  users rating  $M$  items. The (potentially sparse) matrix of ratings  $\mathbf{R} = (R_{ij})_{ij}$ , where  $R_{ij}$  corresponds to the rating of user  $i$  to item  $j$ , is modeled as

$$\begin{aligned} R_{ij} \mid \mathbf{U}, \mathbf{V}, \alpha, I_{ij} = 1 &\sim \mathcal{N}(\mathbf{U}_i^\top \mathbf{V}_j, \alpha^{-1}) \\ \mathbf{U}_i \mid \mu_{\mathbf{U}}, \Lambda_{\mathbf{U}} &\sim \mathcal{N}_p(\mu_{\mathbf{U}}, \Lambda_{\mathbf{U}}^{-1}) \quad i = 1, \dots, U \\ \mathbf{V}_j \mid \mu_{\mathbf{V}}, \Lambda_{\mathbf{V}} &\sim \mathcal{N}_p(\mu_{\mathbf{V}}, \Lambda_{\mathbf{V}}^{-1}) \quad j = 1, \dots, M \end{aligned}$$

where  $\mathbf{U} \in \mathbb{R}^{U \times p}$ ,  $\mathbf{V} \in \mathbb{R}^{M \times p}$  and  $I_{ij} = 1$  if user  $i$  has rated item  $j$ . We adopt the hyperprior

$$\mu_{\mathbf{Z}} \mid \mu_0, \Lambda_{\mathbf{Z}} \sim \mathcal{N}_p(\mu_0, \Lambda_{\mathbf{Z}}^{-1})$$

where  $\Lambda_{\mathbf{Z}} = \text{diag}(\lambda_{\mathbf{z},1}, \dots, \lambda_{\mathbf{z},p})$ ,  $\lambda_{\mathbf{z},j} \sim \Gamma(a_0, b_0)$  for  $j = 1, \dots, p$  and  $\mathbf{Z} = (\mathbf{U}, \mathbf{V})$ . We consider the MovieLens dataset containing  $10^5$  ratings (taking values in  $\{1, 2, 3, 4, 5\}$ ) of 1000 users on 1700 movies. Taking  $p = 20$  and applying a 80% – 20% train-test split, we obtain a 54080-dimensional target posterior distribution  $p(\mathbf{U}, \mathbf{V}, \mu_{\mathbf{U}}, \mu_{\mathbf{V}}, \Lambda_{\mathbf{U}}, \Lambda_{\mathbf{V}} \mid \mathbf{R})$ . We set  $\alpha = 3$ ,  $\mu_0 = 0$ ,  $a_0 = 1$  and  $b_0 = 5$  and use a mini batch size of  $n = N/100 = 800$ .

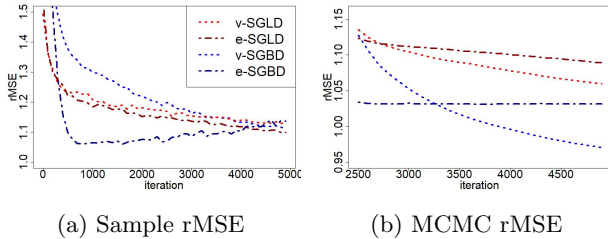
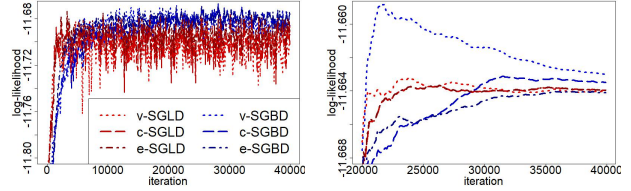


Figure 4. Bayesian Probabilistic Matrix Factorization: Predictive Accuracy on the MovieLens dataset. Sample (left) and MCMC (right) estimates rMSE. Red refers to SGLD and blue to the SGBD. Lighter and dotted lines refer to vanilla implementations of the algorithm, darker and dashed-dotted lines to their extreme variants.

Figure 4 compares v-SGBD, e-SGBD, v-SGLD and e-SGLD in terms of predictive performances, measured in root mean squared errors (rMSE; see supplement for explicit definition). We consider both predictions obtained using single MCMC samples at a given iteration, as well as ones obtained using MCMC ergodic averages. In general, SGBD outperforms SGLD in all variants. e-SGBD converges faster and single samples have a better predictive accuracy. However v-SGBD ergodic average estimates have the best overall predictive accuracy. For all schemes, the step-size maximising predictive performances was selected over a grid.



(a) Samples log-likelihood (b) MCMC log-likelihood

Figure 5. ICA: log likelihood on the MEG dataset. Log likelihood produced by each sample (left) and by the MCMC estimates (right) on held-out data. Red refers to SGLD and blue to SGBD. For both algorithms, the vanilla (lighter dotted lines), corrected (medium scale dashed lines) and extreme (darker dotted-dashed lines) versions are displayed.

### 4.4 Independent Component Analysis

We consider an independent component analysis (Amari et al., 1995; Welling and Teh, 2011) model, where the likelihood of each datapoint  $x_i = (x_{ij})_{j=1, \dots, p} \in \mathbb{R}^p$  for  $i = 1, \dots, N$  is

$$p(x_i \mid W) \propto |\det W| \prod_{k=1}^p \cosh(0.5 \sum_{j=1}^p w_{kj} x_{ij})^{-2}$$

and each entry of  $W = (w_{ij})_{i,j=1, \dots, p}$  is assigned a standard normal prior, i.e.  $w_{ij} \stackrel{iid}{\sim} N(0, 1)$ . We apply the model to the MEG data collected by the Brain Research Unit, from the Helsinki University of Technology, which contains  $N = 17730$  data points of dimensionality 100. To perform our experiments, we extract the first 10 channels (i.e.  $p = 10$ ) and use SG-MCMC to sample from the resulting 100-dimensional posterior distribution  $p(W \mid (x_i)_{i=1, \dots, n})$ . We perform a 80% – 20% train-test split and compare log-likelihood on the unseen test set. We run the samplers for  $T = 4 \times 10^4$  iterations choosing a batch-size of  $n = 100$ . We test the performance on the unseen test set both by computing the log-likelihood produced by each sample, as well as using the one produced by the ergodic averages of  $W$ , see the supplement for explicit definitions. Results are reported in Figure 5. In this example optimal performances were obtained with very small step-sizes, both for SGLD and SGBD, and in such setting the difference between the two algorithms is more limited. Overall, the c-SGBD estimates produce higher levels of log-likelihood on unseen data.

## 5 CONCLUSION

In this paper, we extended the Barker proposal to the stochastic gradient setting, leading to the SGBD algo-



rithm. We studied the bias induced by stochastic gradient noise in the Barker proposal and develop strategies to remove it, for small noise levels, or minimize it, for larger ones. We then compared SGBD to SGLD numerically on simulated and real datasets. Results suggest that, while the two algorithms have similar performances for small step-sizes, SGBD is more robust to hyperparameters choice (thus potentially allowing for larger values to be used in practice) and to heterogeneity in the target gradients (arising from e.g. skewness or ill-conditioning). Overall, SGBD represents a valid alternative to SGLD with minimal algorithmic change and appealing robustness.

There are many directions for future research, including: adding momentum to improve efficiency (similarly to, e.g., SGHMC (Chen et al., 2014)); developing adaptive variants that optimally tunes the step-size across iterations; characterizing more explicitly how the invariant distribution is affected by the choice of the step-size; directly studying impact on predictive accuracy; further exploring the connection with optimization schemes as well as the use of SGBD for optimization purposes.

## Acknowledgements

GZ acknowledges support from the European Research Council (ERC), through StG “PrSc-HDBayLe” grant ID 101076564. LM was partially supported by the National Institutes of Health, grant ID R01ES035625.

## References

- Shun-ichi Amari, Andrzej Cichocki, and Howard Yang. A new learning algorithm for blind signal separation. In D. Touretzky, M.C. Mozer, and M. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8. MIT Press, 1995.
- Adelchi Azzalini. *The skew-normal and related families*, volume 3. Cambridge University Press, 2013.
- Rémi Bardenet, Arnaud Doucet, and Chris Holmes. Towards scaling up markov chain monte carlo: an adaptive subsampling approach. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 405–413, Beijing, China, 22–24 Jun 2014. PMLR.
- Michael Betancourt. The fundamental incompatibility of scalable hamiltonian monte carlo and naive data subsampling. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 533–540, Lille, France, 07–09 Jul 2015. PMLR.
- Shannon Bowling, Mohammad Khasawneh, Sittichai Kaewkuekool, and Byung Cho. A logistic approximation to the cumulative normal distribution. *Journal of Industrial Engineering and Management*, 2, 07 2009.
- Nicolas Brosse, Alain Durmus, and Eric Moulines. The promises and pitfalls of stochastic gradient langevin dynamics. *Advances in Neural Information Processing Systems*, 31, 2018.
- Bob Carpenter, Andrew Gelman, Matthew Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1), 2017.
- Tianqi Chen, Emily Fox, and Carlos Guestrin. Stochastic gradient hamiltonian monte carlo. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1683–1691, Beijing, China, Jun 2014. PMLR.
- Jeremie Coullon, Leah South, and Christopher Nemeth. Efficient and generalizable tuning strategies for stochastic gradient mcmc. *Statistics and Computing*, 33, 04 2023.
- Nan Ding, Youhan Fang, Ryan Babbush, Changyou Chen, Robert D Skeel, and Hartmut Neven. Bayesian sampling using stochastic gradient thermostats. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.
- Simon Duane, A.D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics Letters B*, 195(2):216–222, 1987.
- John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(61):2121–2159, 2011.
- Max Hird, Samuel Livingstone, and Giacomo Zanella. A fresh take on ‘barker dynamics’ for mcmc. In *Monte Carlo and Quasi-Monte Carlo Methods*, 2020.
- Wittawat Jitkrittum, Wenkai Xu, Zoltan Szabo, Kenji Fukumizu, and Arthur Gretton. A linear-time kernel goodness-of-fit test. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.
- Anoop Korattikara, Yutian Chen, and Max Welling. Austerity in mcmc land: Cutting the metropolis-hastings budget. In Eric P. Xing and Tony Jebara,

- editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 181–189, Beijing, China, Jun 2014. PMLR.
- Samuel Livingstone and Giacomo Zanella. The barker proposal: Combining robustness and efficiency in gradient-based mcmc. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2022.
- Xiaoyu Lu, Valerio Perrone, Leonard Hasencllever, Yee Whye Teh, and Sebastian Vollmer. Relativistic Monte Carlo . In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 1236–1245. PMLR, 20–22 Apr 2017.
- Yi-An Ma, Tianqi Chen, and Emily B. Fox. A complete recipe for stochastic gradient mcmc. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS’15, page 2917–2925, Cambridge, MA, USA, 2015. MIT Press.
- Radford Neal. Mcmc using hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2012.
- Christopher Nemeth and Paul Fearnhead. Stochastic gradient markov chain monte carlo. *Journal of the American Statistical Association*, 116(533):433–450, 2021.
- Samuel Power and Jacob Vorstrup Goldman. Accelerated sampling on discrete spaces with non-reversible markov processes. *arXiv preprint arXiv:1912.04681*, 2019.
- Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400 – 407, 1951.
- Gareth O. Roberts and Jeffrey S. Rosenthal. Optimal scaling of discrete approximations to langevin diffusions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(1):255–268, 1995.
- Gareth O Roberts and Richard L Tweedie. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, pages 341–363, 1996.
- Ruslan Salakhutdinov and Andriy Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th International Conference on Machine Learning*, ICML ’08, page 880–887, New York, NY, USA, 2008. Association for Computing Machinery.
- Haoran Sun, Hanjun Dai, Bo Dai, Haomin Zhou, and Dale Schuurmans. Discrete langevin samplers via wasserstein gradient flow. In *International Conference on Artificial Intelligence and Statistics*, pages 6290–6313. PMLR, 2023.
- Yee Whye Teh, Alexandre H. Thiery, and Sebastian J. Vollmer. Consistency and fluctuations for stochastic gradient langevin dynamics. *Journal of Machine Learning Research*, 17(1):193–225, 2016.
- Jure Vogrinc, Samuel Livingstone, and Giacomo Zanella. Optimal design of the Barker proposal and other locally balanced Metropolis–Hastings algorithms. *Biometrika*, 10 2022. ISSN 1464-3510.
- Sebastian J. Vollmer, Konstantinos C. Zygalakis, and Yee Whye Teh. Exploration of the (non-)asymptotic bias and variance of stochastic gradient langevin dynamics. *Journal of Machine Learning Research*, 17 (159):1–48, 2016.
- 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, page 681–688, Madison, WI, USA, 2011. Omnipress.
- Giacomo Zanella. Informed proposals for local mcmc in discrete spaces. *Journal of the American Statistical Association*, 115(530):852–865, 2020.

## Checklist

- For all models and algorithms presented, check if you include:
  - A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
  - An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
- For any theoretical claim, check if you include:
  - Statements of the full set of assumptions of all theoretical results. [Yes]
  - Complete proofs of all theoretical results. [Yes]
  - Clear explanations of any assumptions. [Yes]
- For all figures and tables that present empirical results, check if you include:
  - The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
  - All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]

- (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
  - (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
- (a) Citations of the creator If your work uses existing assets. [Yes]
  - (b) The license information of the assets, if applicable. [Yes]
  - (c) New assets either in the supplemental material or as a URL, if applicable. [Yes]
  - (d) Information about consent from data providers/curators. [Not Applicable]
  - (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]
5. If you used crowdsourcing or conducted research with human subjects, check if you include:
- (a) The full text of instructions given to participants and screenshots. [Not Applicable]
  - (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
  - (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

---

## Supplementary Materials

---

### A ALGORITHMS

In this section we provide the pseudocode for the e-SGBD, v-SGLD and c-SGLD algorithms described in Section 3.4 of the paper.

---

#### Algorithm 4: Extreme Stochastic Gradient Barker Dynamics (e-SGBD)

---

**Input:**  $\theta^{(0)} \in \mathbb{R}^d, \sigma > 0$

```

1 for  $t=1, \dots, T$  do
2   Draw  $\mathcal{S}_n \subset \{1, \dots, N\}$  uniformly at random;
3   for  $j=1, \dots, d$  ;                                     // Can be parallelized
4     do
5       Compute  $\hat{\partial}_j g(\theta^{(t-1)})$  as in (3);
6       Draw  $w_j^{(t)} \sim N(\sigma, (0.1\sigma)^2)$ ;
7       Set  $b_j^{(t)} = 1$  if  $w_j^{(t)} \hat{\partial}_j g(\theta^{(t-1)}) > 0$  otherwise  $b_j^{(t)} = -1$ ;
8       Update  $\theta^{(t)} \leftarrow \theta^{(t-1)} + b_j^{(t)} w_j^{(t)}$  ;
9     end
10 end

```

---



---

#### Algorithm 5: Stochastic Gradient Langevin Dynamics (v-SGLD)

---

**Input:**  $\theta^{(0)} \in \mathbb{R}^d, \sigma > 0$

```

1 for  $t=1, \dots, T$  do
2   Draw  $\mathcal{S}_n \subset \{1, \dots, N\}$  uniformly at random;
3   for  $j=1, \dots, d$  ;                                     // Can be parallelized
4     do
5       Compute  $\hat{\partial}_j g(\theta^{(t-1)})$  as in (3);
6       Draw  $\epsilon_j^{(t)} \sim N(0, \sigma^2)$ ;
7       Update  $\theta^{(t)} \leftarrow \theta^{(t-1)} + \frac{\sigma^2}{2} \hat{\partial}_j g(\theta^{(t-1)}) + \epsilon_j^{(t)}$ ;
8     end
9 end

```

---

---

**Algorithm 6:** Corrected Stochastic Gradient Langevin Dynamics (c-SGLD)

---

**Input:**  $\theta^{(0)} \in \mathbb{R}^d, \sigma > 0, \beta \in (0, 1), \{\hat{\tau}_j^{(0)}\}_{j=1, \dots, d}$

```

1 for  $t=1, \dots, T$  do
2   Draw  $\mathcal{S}_n \subset \{1, \dots, N\}$  uniformly at random;
3   for  $t=1, \dots, T$ ; // Can be parallelized
4   do
5     Compute  $\hat{\partial}_j g(\theta^{(t-1)})$  as in (3);
6     Update  $\hat{\tau}_j^{(t)} \leftarrow (1 - \beta)\hat{\tau}_j^{(t-1)} + \beta \sqrt{\frac{1}{n-1} \sum_{i \in \mathcal{S}_n} (\partial_j g_i(\theta^{(t-1)}) - \frac{1}{n} \sum_{i \in \mathcal{S}_n} \partial_j g_i(\theta^{(t-1)}))^2}$ ;
7     Draw  $\epsilon_j^{(t)} \sim N\left(0, \max\left(0, \sigma^2 - \frac{\hat{\tau}_j^{(t)2}}{4} \sigma^4\right)\right)$ ;
8     Update  $\theta^{(t)} \leftarrow \theta^{(t-1)} + \frac{\sigma^2}{2} \hat{\partial}_j g(\theta^{(t-1)}) + \epsilon_j^{(t)}$ ;
9   end
10 end

```

---

## B PROOFS AND ADDITIONAL LEMMAS

### B.1 Proof of Proposition 1

We first state two auxiliary lemmas.

**Lemma 1.** *Let  $(\delta, z) \in \mathbb{R}^2$  and  $\eta \neq 0$ . Then  $\frac{1}{2}(p(\delta - \eta, z) + p(\delta + \eta, z)) \leq p(\delta, z)$  if and only if  $z\delta \geq 0$ , with equality only if  $z\delta = 0$ .*

*Proof.* We have

$$2p(\delta, z) - (p(\delta - \eta, z) + p(\delta + \eta, z)) = (1 + e^{-z\delta})^{-1} - \left( (1 + e^{-z\delta+z\eta})^{-1} + (1 + e^{-z\delta-z\eta})^{-1} \right). \quad (15)$$

Define  $c = e^{-z\delta}$ . Expanding the right hand side of (15), we obtain

$$(1 + c)^{-1} - \left( (1 + ce^{+z\eta})^{-1} + (1 + ce^{-z\eta})^{-1} \right) = \frac{c(1 - c)((e^{-z\eta} + e^{z\eta}) - 2)}{(1 + c)(1 + ce^{-z\eta})(1 + ce^{+z\eta})} \quad (16)$$

Note that the denominator on the right hand side of (16) is strictly positive and  $(e^{-z\eta} + e^{z\eta}) - 2 \geq 0$  by Jensen's inequality with strict inequality if  $z \neq 0$ . The result follows by noting that  $c > 0$  always, and  $1 - c \geq 0$  if and only if  $z\delta \geq 0$ , with strict inequality if  $z\delta > 0$ .  $\square$

**Lemma 2.** *Let  $(\delta, z) \in \mathbb{R}^2$  and  $\eta \neq 0$ . Then  $\frac{1}{2}(p(\delta - \eta, z) + p(\delta + \eta, z)) \geq 0.5$  if and only if  $z\delta \geq 0$ , with equality only if  $z\delta = 0$ .*

*Proof.* Define  $c := e^{-z\delta}$  and consider the following:

$$\begin{aligned} \frac{1}{2}(p(\delta + \eta, z) + p(\delta - \eta, z)) &= \frac{1}{2} \left( (1 + ce^{-z\eta})^{-1} + (1 + ce^{z\eta})^{-1} \right) \\ &= \frac{2 + c(e^{-z\eta} + e^{z\eta})}{2(1 + ce^{-z\eta})(1 + ce^{z\eta})} \\ &= \frac{2 + c(e^{-z\eta} + e^{z\eta})}{2 + 2c(e^{-z\eta} + e^{z\eta}) + c^2} \end{aligned} \quad (17)$$

The result follows by noting that  $c^2 \leq 1$  if and only if  $z\delta \geq 0$ , with strict inequality if  $z\delta > 0$ , and  $(e^{-z\eta} + e^{z\eta}) - 2 \geq 0$  by Jensen's inequality with strict inequality if  $z \neq 0$ .  $\square$

We can now prove Proposition 1.

*Proof of Proposition 1.* Consider first the case  $z\partial_j g(\theta) > 0$  and  $\tau_\theta > 0$ . Denoting the distribution of  $\eta_\theta$  by  $P_{\eta_\theta}$ , we have

$$\begin{aligned} \mathbb{E} [p(\partial_j \hat{g}(\theta), z)] &\stackrel{(I)}{=} \int_0^{+\infty} (p(\partial_j g(\theta) + \eta_\theta, z) + p(\partial_j g(\theta) - \eta_\theta, z)) dP_{\eta_\theta}(\eta_\theta) \\ &= \int_0^{+\infty} \left( \left(1 + e^{-z(\partial_j g(\theta) + \eta_\theta)}\right)^{-1} + \left(1 + e^{-z(\partial_j g(\theta) - \eta_\theta)}\right)^{-1} \right) dP_{\eta_\theta}(\eta_\theta) \\ &\stackrel{(II)}{\leq} \int_0^{+\infty} 2 \left(1 + e^{-z(\partial_j g(\theta))}\right)^{-1} dP_{\eta_\theta}(\eta_\theta) \\ &\stackrel{(III)}{=} \int_{-\infty}^{+\infty} \left(1 + e^{-z(\partial_j g(\theta))}\right)^{-1} dP_{\eta_\theta}(\eta_\theta) = p(\partial_j g(\theta), z), \end{aligned} \quad (18)$$

where (I) and (III) follow from the symmetry assumption in Condition 1, and (II) follows from Lemma 1. Moreover,  $z\partial_j g(\theta) > 0$  implies  $p(\partial_j g(\theta), z) > 0.5$  and  $\frac{1}{2}(p(\partial_j g(\theta) + \eta_\theta, z) + p(\partial_j g(\theta) - \eta_\theta, z)) > 0.5$  for all values  $\eta_\theta \neq 0$  by Lemma 2. It follows, again using the symmetry assumption in Condition 1, that

$$0.5 < \mathbb{E} \left[ \hat{p}(\hat{\partial}_j g(\theta), z) \right] < p(\partial_j g(\theta), z), \quad (19)$$

Using the same argument, it is easy to show that the reverse inequalities hold if  $z\partial_j g(\theta) < 0$ , leading to  $|p(\partial_j g(\theta), z) - 0.5| > \left| \mathbb{E} \left[ \hat{p}(\hat{\partial}_j g(\theta), z) \right] - 0.5 \right|$  as desired. The argument for the case  $z\partial_j g(\theta) < 0$  is analogous. Finally, if  $z\partial_j g(\theta) = 0$  or  $\tau_\theta = 0$ , we have  $\mathbb{E} [p(\partial_j \hat{g}(\theta), z)] = p(\partial_j g(\theta), z) = 0.5$ .  $\square$

## B.2 Proof of Proposition 2

*Proof.* Let  $F(x) = (1 + \exp(-x))^{-1}$  be the CDF of the logistic distribution and  $\Phi$  the one of the standard Normal distribution. Then we have  $|F(x) - \Phi(x/1.702)| < 0.0095$  for all  $x \in \mathbb{R}$ , see e.g. Bowling et al. (2009, Section 3.2). By definition of  $p$ , this implies

$$\left| p(\delta, z) - \Phi \left( \frac{z\delta}{1.702} \right) \right| < 0.0095 \quad \delta, z \in \mathbb{R} \quad (20)$$

and, as a result,

$$\left| \mathbb{E} \left[ p(\hat{\partial}_j g(\theta), z) - \Phi \left( \frac{z\hat{\partial}_j g(\theta)}{1.702} \right) \right] \right| < 0.0095. \quad (21)$$

Hence,  $\left| \mathbb{E} \left[ p(\hat{\partial}_j g, z) \right] - \Phi \left( \frac{z\partial_j g(\theta)}{\sqrt{1.702^2 + z^2\tau_\theta^2}} \right) \right| < 0.0095$  noting that

$$\mathbb{E} \left[ \Phi \left( \frac{z\hat{\partial}_j g(\theta)}{1.702} \right) \right] = \Phi \left( \frac{z\partial_j g(\theta)}{\sqrt{1.702^2 + z^2\tau_\theta^2}} \right), \quad (22)$$

which follows by  $\hat{\partial}_j g(\theta) \sim \mathcal{N}(\partial_j g(\theta), \tau_\theta^2)$  and standard properties of the normal distribution. The result in Equation (7) follows using the bound (20) with an application of the triangle inequality.  $\square$

## B.3 Proof of Corollary 1

*Proof.* Consider first the case when  $\tau_\theta < 1.702/|z|$ . Then, replacing  $z$  with  $z\alpha$  in Proposition 2, we obtain

$$\left| \mathbb{E} \left[ p(\hat{\partial}_j g, \alpha z) \right] - p \left( \frac{1.702\alpha}{\sqrt{1.702^2 + z^2\tau_\theta^2\alpha^2}} \partial_j g(\theta), z \right) \right| < 0.019. \quad (23)$$

Taking  $\alpha = \frac{1.702}{\sqrt{1.702^2 - \tau_\theta^2 z^2}}$ , we have  $p(\hat{\partial}_j g, \alpha z) = \tilde{p}(\hat{\partial}_j g, z)$  and the second term in the absolute value on the left hand side of (23) simplifies to  $p(\partial_j g(\theta), z)$  leading to the desired result.

Consider now the case when  $\bar{\tau}(\partial_j g(\theta), z) \geq 1.702/|z|$  and  $\tau_\theta \in [1.702/|z|, \bar{\tau}(\partial_j g(\theta), z)]$ . In this case, we have  $\bar{p}(\hat{\partial}_j g(\theta), z) = \mathbf{1}(\hat{\partial}_j g(\theta)z > 0)$ . We first consider the case when  $z\partial_j g(\theta) > 0$ . Under Condition 2, we have

$$\mathbb{E} \left[ \bar{p} \left( \hat{\partial}_j g, z \right) \right] = \Phi(|\partial_j g(\theta)|/\tau_\theta). \quad (24)$$

The right hand side of (24) is a strictly increasing function of  $\tau_\theta$ , hence it can be lower and upper bounded by

$$\Phi(|\partial_j g(\theta)|/\bar{\tau}(\partial_j g(\theta), z)) \leq \Phi(|\partial_j g(\theta)|/\tau_\theta) \leq \Phi(\partial_j g(\theta)z/1.702). \quad (25)$$

Define  $D(x) := \Phi(x/1.702) - (1 + e^{-x})^{-1}$  and expand the upper bound in (25) as

$$\begin{aligned} \Phi(\partial_j g(\theta)z/1.702) &= \left(1 + e^{-\partial_j g(\theta)z}\right) - D(\partial_j g(\theta)z) \\ &\leq p(\partial_j g(\theta), z) + 0.0095. \end{aligned} \quad (26)$$

Moreover, note that, when  $z\partial_j g(\theta) > 0$ , the left hand side in (24) is equal to  $p(\partial_j g(\theta), z)$  by the definition of  $\tau(\partial_j g(\theta), z)$ , i.e. where the inequality follows from the definition of  $p$  and  $|D(x)| < 0.0095$  for all  $x \in \mathbb{R}$  (Bowling et al., 2009, Section 3.2).

$$\Phi(\partial_j g(\theta)/\bar{\tau}(\partial_j g(\theta), z)) = p(\partial_j g(\theta), z). \quad (27)$$

Thus, combining all of the above, we get

$$p(\partial_j g(\theta), z) \leq \mathbb{E} \left[ \bar{p} \left( \hat{\partial}_j g(\theta), z \right) \right] \leq p(\partial_j g(\theta), z) + 0.0095. \quad (28)$$

When  $\partial_j g(\theta)z \leq 0$ , it is easy to show that the reverse inequalities hold, leading to the following bound:

$$p(\partial_j g(\theta), z) - 0.0095 \leq \mathbb{E} \left[ \bar{p} \left( \hat{\partial}_j g, z \right) \right] \leq p(\partial_j g(\theta), z). \quad (29)$$

□

#### B.4 Proof of Proposition 3

*Proof.* To prove the statement, consider the expectation of the *extreme*-estimator defined in Section 3.4 of the paper,  $\bar{p}$ ,

$$\mathbb{E} \left[ \bar{p} \left( \hat{\partial}_j g(\theta), z \right) \right] = \Phi(\tau_\theta^{-1} |\partial_j g(\theta)| \operatorname{sgn}(z\partial_j g(\theta))). \quad (30)$$

Note that the definition  $\bar{\tau}(\partial_j g(\theta), z)$  in (9) implies that

$$\Phi(|\partial_j g(\theta)| \operatorname{sgn}(z\partial_j g(\theta))/\tau(\partial_j g(\theta), z)) = p(\partial_j g(\theta), z). \quad (31)$$

Consider the case when  $\tau_\theta > \tau^*$ . Since  $\bar{\tau}(\delta, z)$  is a continuous function of  $\delta$ , there exists  $\bar{\delta} \in \mathbb{R}$  such that  $\tau^* < \bar{\tau}(\bar{\delta}, z) < \tau_\theta$ . Notice that the right hand side of (30) is decreasing in  $\tau_\theta$  if  $z\partial_j g(\theta) > 0$  and increasing otherwise. Hence, for a value of the gradient  $\bar{\delta}$  and noise standard deviation  $\tau_\theta > \bar{\tau}(\bar{\delta}, z)$ , we obtain

$$|p(\bar{\delta}, z) - 0.5| < \left| \mathbb{E} \left[ \bar{p} \left( \hat{\partial}_j g(\theta), z \right) \right] - 0.5 \right|. \quad (32)$$

By Proposition 4, we know that, for any other symmetric estimator  $\hat{p}$  satisfying (12), we have

$$\left| \mathbb{E} \left[ \bar{p} \left( \hat{\partial}_j g(\theta), z \right) \right] - 0.5 \right| < \left| \mathbb{E} \left[ \hat{p} \left( \hat{\partial}_j g(\theta), z \right) \right] - 0.5 \right| \quad (33)$$

Hence, no unbiased symmetric estimator exists for  $\tau_\theta > \tau^*$ .

Finally, note that the absence of any unbiased symmetric estimator implies the absence of any unbiased estimator, since any unbiased estimator needs to be symmetric. Indeed, assume by contradiction that  $\hat{p}(\hat{\delta}, z)$  is unbiased for  $p(\delta, z)$  when  $\hat{\delta} \sim \mathcal{N}(\delta, \tau_\theta)$  but it is not symmetric, i.e.  $\hat{p}(\hat{\delta}, z) + \hat{p}(-\hat{\delta}, z) \neq 1$  for some  $\hat{\delta} \in \mathbb{R}$ . Consider  $z$  fixed and define  $h(\hat{\delta}) = \hat{p}(\hat{\delta}, z) + \hat{p}(-\hat{\delta}, z) - 1$ . Unbiasedness imply  $\mathbb{E} \left[ \hat{p}(\hat{\delta}, z) + \hat{p}(-\hat{\delta}, z) \right] = p(\delta, z) + p(-\delta, z) = 1$ , or equivalently  $\mathbb{E} \left[ h(\hat{\delta}) \right] = 0$  with  $\hat{\delta} \sim \mathcal{N}(\delta, \tau_\theta^2)$ , for all  $\delta \in \mathbb{R}$ . Since  $\hat{\delta}$  is a complete sufficient statistics for  $\hat{\delta} \sim \mathcal{N}(\delta, \tau_\theta^2)$ , the ladder condition implies  $h(\hat{\delta}) \equiv 0$ , determining a contradiction. □

### B.5 Proof of Proposition 4

*Proof.* The case  $z = 0$  is trivial. We prove the result when  $z > 0$ , the case  $z < 0$  is analogous. For any symmetric estimator  $\hat{p}$  satisfying (12), we have

$$\begin{aligned}
 \mathbb{E} \left[ \hat{p}(\hat{\partial}_j g(\theta), z) \right] &= \int_{-\infty}^{+\infty} \hat{p}(\partial_j g(\theta) + \eta_\theta, z) f_\theta(\eta_\theta) d\eta_\theta \\
 &\stackrel{(I)}{=} \int_{-\infty}^{+\infty} \hat{p}(\varepsilon, z) f_\theta(\varepsilon - \partial_j g(\theta)) d\varepsilon \\
 &= \int_0^{+\infty} \hat{p}(\varepsilon, z) f_\theta(\varepsilon - \partial_j g(\theta)) d\varepsilon + \int_{-\infty}^0 \hat{p}(\varepsilon, z) f_\theta(\varepsilon - \partial_j g(\theta)) d\varepsilon \\
 &\stackrel{(II)}{=} \int_0^{+\infty} \hat{p}(\varepsilon, z) f_\theta(\varepsilon - \partial_j g(\theta)) d\varepsilon + \int_0^{+\infty} \hat{p}(-\varepsilon, z) f_\theta(-\varepsilon - \partial_j g(\theta)) d\varepsilon \\
 &\stackrel{(III)}{=} \int_0^{+\infty} \hat{p}(\varepsilon, z) f_\theta(\varepsilon - \partial_j g(\theta)) d\varepsilon + \int_0^{+\infty} \hat{p}(-\varepsilon, z) f_\theta(\varepsilon + \partial_j g(\theta)) d\varepsilon
 \end{aligned} \tag{34}$$

where (I) and (II) follows by changes of variables ( $\varepsilon := \eta_\theta + \partial_j g(\theta)$  and  $\varepsilon := -\varepsilon$ , respectively), and (III) follows by Conditions 1 and 3 which implies  $f_\theta(x) = f_\theta(-x)$  for all  $x \in \mathbb{R}$ .

The expected value of the *extreme*-estimator  $\bar{p}$  is given by:

$$\begin{aligned}
 \mathbb{E} \left[ \bar{p}(\hat{\partial}_j g(\theta), z) \right] &= \int_{-\infty}^{+\infty} \bar{p}(\partial_j g(\theta) + \eta_\theta, z) f_\theta(\eta_\theta) d\eta_\theta \\
 &= \int_{-\partial_j g(\theta)}^{+\infty} 1 f_\theta(\eta_\theta) d\eta_\theta \\
 &\stackrel{(I)}{=} \int_0^{+\infty} 1 f_\theta(\varepsilon - \partial_j g(\theta)) d\varepsilon \\
 &\stackrel{(II)}{=} \int_0^{+\infty} \hat{p}(\varepsilon, z) f_\theta(\varepsilon - \partial_j g(\theta)) d\varepsilon + \int_0^{+\infty} \hat{p}(-\varepsilon, z) f_\theta(\varepsilon - \partial_j g(\theta)) d\varepsilon
 \end{aligned} \tag{35}$$

where (I) follows from the change of variable  $\varepsilon := \eta_\theta + \partial_j g(\theta)$  and (II) follows from (12). Thus

$$\mathbb{E} \left[ \bar{p}(\hat{\partial}_j g(\theta), z) \right] - \mathbb{E} \left[ \hat{p}(\hat{\partial}_j g(\theta), z) \right] = \int_0^{+\infty} \hat{p}(-\varepsilon, z) (f_\theta(\varepsilon - \partial_j g(\theta)) - f_\theta(\varepsilon + \partial_j g(\theta))) d\varepsilon. \tag{36}$$

Moreover, note that by Conditions 1 and 3, if  $\partial_j g(\theta) > 0$  we have  $f_\theta(\varepsilon + \partial_j g(\theta)) < f_\theta(\varepsilon - \partial_j g(\theta))$ , while if  $\partial_j g(\theta) < 0$  we have  $f_\theta(\varepsilon + \partial_j g(\theta)) > f_\theta(\varepsilon - \partial_j g(\theta))$ , for every  $\varepsilon > 0$ . Hence, the right hand side in equation (36) is greater than 0 if  $\partial_j g(\theta) > 0$  and smaller than 0 otherwise, with equality holding if and only if  $\hat{p}(\varepsilon, z)$  for all  $\varepsilon < 0$ , which holds if and only if  $\hat{p} = \bar{p}$ , if we exclude the trivial estimator  $\hat{p} \equiv 1$ .  $\square$

### B.6 Proof of Corollary 2

*Proof.* By Proposition 4, we have  $\mathbb{E} \left[ \bar{p}(\hat{\partial}_j g(\theta), z) \right] > \mathbb{E} \left[ \hat{p}(\hat{\partial}_j g(\theta), z) \right]$  if  $\partial_j g(\theta)z > 0$ , while the reverse inequality holds if  $\partial_j g(\theta)z < 0$ . Moreover, by (30), when  $\tau_\theta > \bar{\tau}(\partial_j g(\theta), z)$  the expectation of  $\bar{p}(\hat{\partial}_j g(\theta), z)$  is shrunk to 0.5 compared to  $p(\partial_j g(\theta), z)$ . Thus, for  $\partial_j g(\theta)z > 0$ , we have

$$0.5 < \mathbb{E} \left[ \hat{p}(\hat{\partial}_j g(\theta), z) \right] < \mathbb{E} \left[ \bar{p}(\hat{\partial}_j g(\theta), z) \right] < p(\partial_j g(\theta), z), \tag{37}$$

while the reverse inequalities hold when  $\partial_j g(\theta)z < 0$ , proving the result.  $\square$

## C CONNECTION BETWEEN SGBD AND SGLD NOISE TOLERANCE

One can compare the noise tolerance of SGBD with the one of SGLD. Consider the recursion of SGLD

$$\theta^{(t+1)} \leftarrow \theta^{(t)} + \frac{\sigma^2}{2} \hat{\nabla} g(\theta^{(t)}) + z \quad z \sim \mathcal{N}(0, \sigma^2). \tag{38}$$



Under Condition 2, this recursion is equivalent to

$$\theta^{(t+1)} \leftarrow \theta^{(t)} + \frac{\sigma^2}{2} \nabla g(\theta^{(t)}) + \tilde{z} \quad \tilde{z} \sim \mathcal{N}(0, \sigma^2 + (\sigma^2/2)^2 \tau_\theta^2). \quad (39)$$

When  $\tau_\theta \leq 2\sigma^{-1}$ , it is possible to correct exactly the recursion reducing the variance of the artificial noise: namely, if  $z \sim \mathcal{N}(0, \sigma^2 - (\sigma^2/2)^2 \tau_\theta^2)$ , we obtain back the exact unadjusted Langevin proposal, i.e.  $\theta^{(t+1)} \sim \mathcal{N}(\theta^{(t)} + \frac{\sigma^2}{2} \nabla g(\theta^{(t)}), \sigma^2)$ . On the contrary, if  $\tau_\theta > 2\sigma^{-1}$ , simple variance arguments show that there is no distribution for the noise  $z$  (assuming it to be independent from  $\eta_\theta$ ) such that the resulting proposal coincides with the exact unadjusted Langevin one. In this sense, the noise tolerance of SGLD under Condition 2 is  $\tau_{SGLD}^* = 2\sigma^{-1}$ . Contrary to the one of SGBD, such value depends directly on the hyperparameter  $\sigma$  and not on the sampled increment  $z$ . However, there are similarities with the one of SGBD identified in Section 3.3 of the paper. In particular, the size of the proposed increment under  $\mu$  is of order  $\sigma$ , think e.g. at the choice  $\mu = 0.5N(-\sigma, (0.1\sigma)^2) + 0.5N(\sigma, (0.1\sigma)^2)$ , meaning that we can roughly interpret the noise tolerance of SGBD under Condition 2 as being  $\tau^* \approx 1.596\sigma^{-1}$ . This value is similar to the one of SGLD and exhibit the same dependance with respect to  $\sigma$ , despite the constant in front being slightly smaller.

This rough analysis suggests that the amount of Gaussian noise the two algorithms can tolerate while being able to exactly recover the original proposal is similar. On the other hand, one can expect that, due to the non-linear use of gradients in SGBD, the two algorithms will exhibit significantly different behavior in the case of noise  $\eta_\theta$  with larger variance and/or an heavier-tail distribution, with SGBD providing a more and stable robust behaviour (e.g. in the sense of resulting in a proposal closer to the original exact unadjusted scheme). We illustrate this phenomenon numerically in this section, leaving a more detailed theoretical analysis of this behaviour to future work. We consider a standard Normal target, i.e.  $\pi(\theta) = \phi(\theta)$  and add to the true gradient symmetric noise with Laplace or Cauchy distribution and scale parameter  $\tau_\theta$ . Compared to the example presented in Section

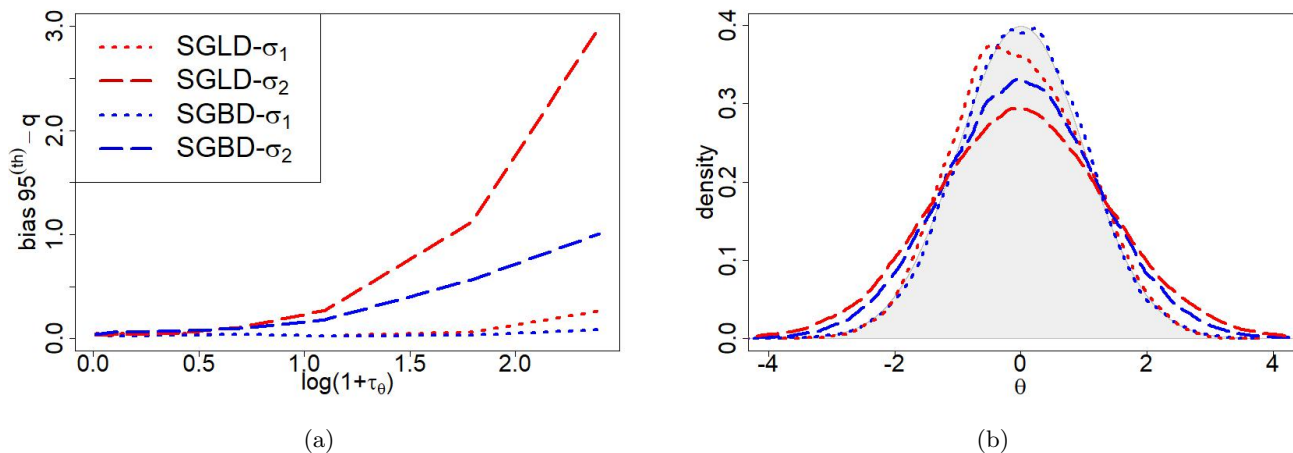


Figure 6. Toy Example: Normal Target with Laplace Noise in gradient estimates. Scale parameter of the noise distribution (in log-scale) versus bias of the 95<sup>th</sup> quantile (left) and estimate of the stationary distributions when  $\tau_\theta = e^{1.5} - 1$  (right). Red refers to v-SGLD and blue to v-SGBD. Dotted (dashed resp.) lines are produced with  $\sigma_1 = 0.1$  ( $\sigma_2 = 0.5$  resp.).

4.1, this setting allows to focus on the robustness to heavy-tailed gradient noise rather than skewness in the target distribution. We consider two values for the step-size  $\sigma$ , namely  $\sigma_1 = 0.1$  and  $\sigma_2 = 0.5$  and run v-SGBD and v-SGLD for  $T = 2 \times 10^5$  discarding the first half as burn-in. Figure 6a shows the increase in the bias of the 95<sup>th</sup> quantile of the invariant distribution relative to  $\pi$ , when  $\eta_\theta \sim \mathbf{Laplace}(0, \tau_\theta)$  as  $\tau_\theta$  increases, and Figure 6b reports the density estimates when  $\tau_\theta = e^{1.5} - 1$ . With Laplace distributed noise, v-SGLD and v-SGBD with a small step-size behave similarly. However, v-SGBD is more robust than v-SGLD when the step-size is increased and exhibits a smaller bias. Figure 7 shows the same plots, where the noise is Cauchy distributed, i.e.  $\eta_\theta \sim \mathbf{Cauchy}(0, \tau_\theta)$ . In this scenario, v-SGBD is dramatically more robust than v-SGLD.

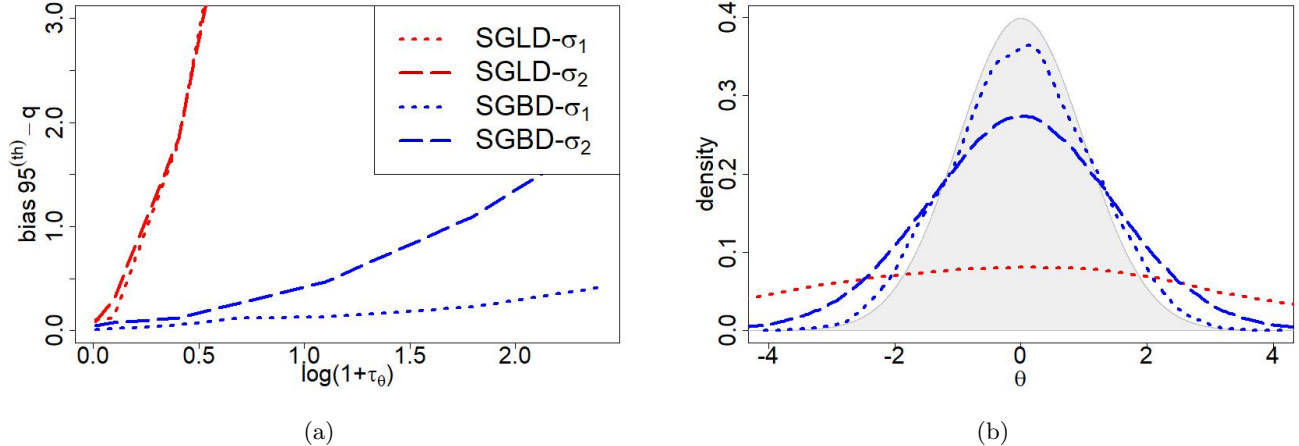


Figure 7. Toy Example: Normal Target with Cauchy Noise in gradient estimates. Scale parameter of the noise distribution (in log-scale) versus bias of the 95<sup>th</sup> quantile (left) and estimate of the stationary distributions when  $\tau_\theta = e^{1.5} - 1$  (right). Red refers to v-SGLD and blue to v-SGBD. Dotted (dashed resp.) lines are produced with  $\sigma_1 = 0.1$  ( $\sigma_2 = 0.5$  resp.). v-SGLD with step-size  $\sigma_2$  fails completely to sample from the target and its density estimate is omitted.

## D ADDITIONAL EXPERIMENTS

We report the additional simulations details and results. All experiments were run on a laptop with 11th Gen Intel(R) Core(TM) i7-1165G7 2.80 GHz using R version 4.3.1.

### D.1 Empirical Simulation Of The Correction for $p$

We present additional details for Figure 1 in the paper. Figure 8 reports the same plot for three additional coordinates. We take the last sample from the v-SGBD chain used to produce the Figure 16a in the paper, we repeatedly subsample a mini-batch, store the gradient for each coordinate, estimate its standard deviation, and compute  $\hat{p}$  and  $\tilde{p}$ . Figure 8 reports the resulting Monte Carlo average of  $\hat{p}$  and  $\tilde{p}$  as the value of the proposed increment varies. The figure was produced with the following values: (a)  $\partial_j g(\theta) = -25.15$ ,  $\tau_\theta = 22.72$ , (b)  $\partial_j g(\theta) = 40.39$ ,  $\tau_\theta = 25.63$ , (c)  $\partial_j g(\theta) = -23.21$ ,  $\tau_\theta = 22.88$  (d)  $\partial_j g(\theta) = -13.03$ ,  $\tau_\theta = 20.27$ .

### D.2 Toy Example: Skew-Normal target with isotropic Gaussian Noise

Figure 9 reports the results obtained using also corrected variants of the algorithms and with a different value for the standard deviation of  $\eta_\theta$  for the experiment presented in Section 4.1 of the paper. With a gradient noise standard deviation equal to the one of the target, we observe little difference between vanilla and corrected variants of the algorithms (Figure 9a). When the standard deviation is large, i.e.  $\tau_\theta = 10 \times sd(\pi_\alpha)$ , corrected variants achieve a lower bias than their vanilla counterparts with a large step-size (Figure 9b). Under all configurations, SGBD displays increased robustness to skewness and to tuning with respect to SGLD.

Next, we perform a sensitivity analysis to the step-size choice. Figure 10 reports the density estimates of the samples obtained via v-SGBD with the step-size respectively equal to  $0.05 \times sd(\pi_\alpha)$ ,  $0.1 \times sd(\pi_\alpha)$ ,  $0.5 \times sd(\pi_\alpha)$ , and  $0.75 \times sd(\pi_\alpha)$ , for  $\alpha = 20$ , where  $sd(\pi_\alpha)$  denotes the standard deviation of the target distribution. v-SGBD appears to be very robust to the step-size choice, and only slightly inflates the scale of the target distribution under the configuration with the largest step-size.

### D.3 Binary Regression With Scale Heterogeneity

Figure 11 shows the density estimates for all coordinates of the posterior samples obtained in the experiment in Section 4.2, and Figure 12 shows the corresponding trace plots.

Figure 13 reports the density estimates for all coordinates of the posterior samples from a second experiment where we tuned the step-sizes via MAMBA (Coullon et al., 2023). MAMBA uses a multi-armed bandits algorithm to minimize the Finite Set Stein Discrepancy (Jitkrittum et al., 2017) between true posterior and its Monte

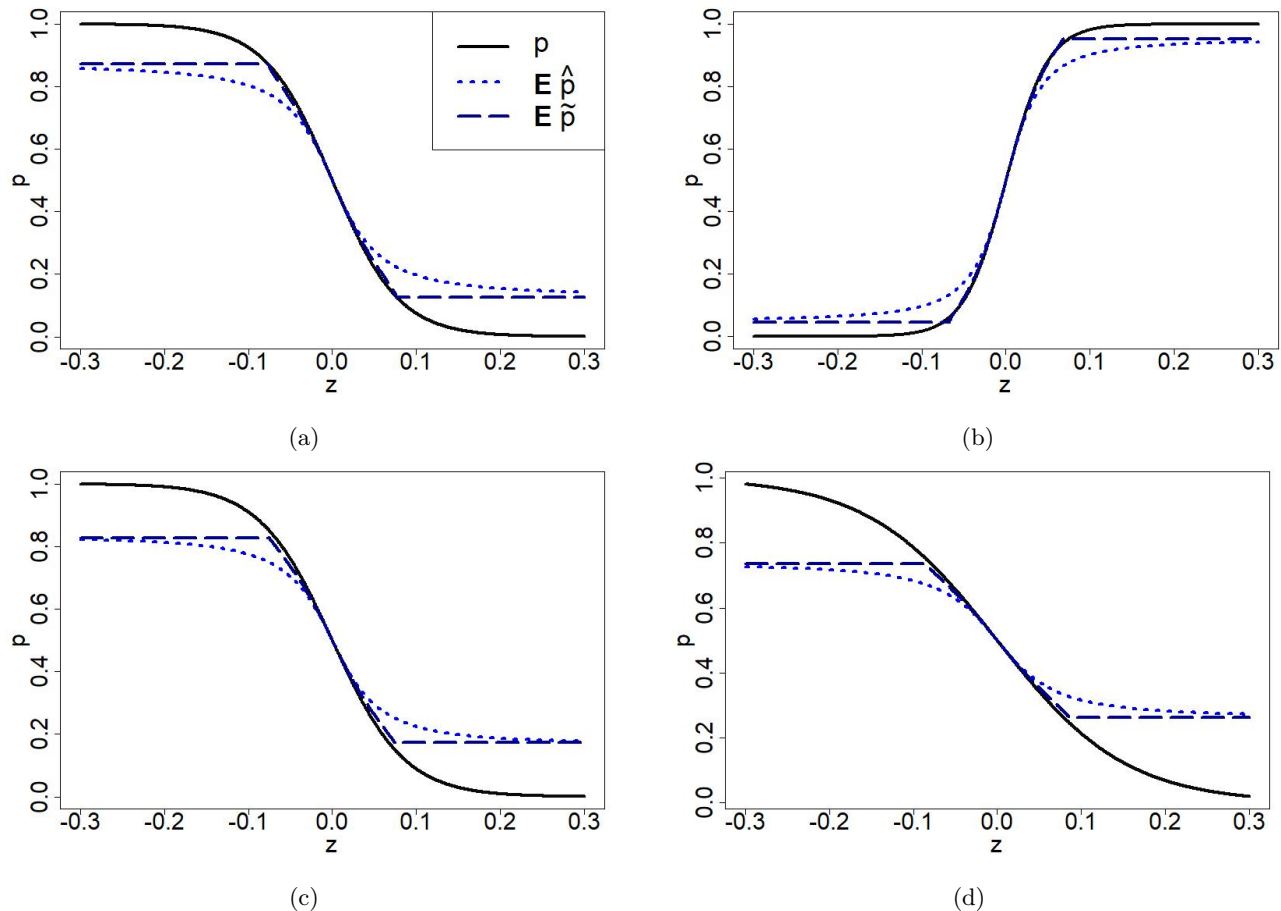


Figure 8. Shrinkage effect and bias correction. Plot of  $p(\partial_j g(\theta), z)$  (black line;  $p$ ) and Monte Carlo estimates of  $\mathbb{E}[p(\hat{\partial}_j g(\theta), z)]$  (dotted blue line;  $\mathbb{E}\hat{p}$ ), and  $\mathbb{E}[\tilde{p}(\hat{\partial}_j g(\theta), z)]$  (dashed dark blue line;  $\mathbb{E}\tilde{p}$ ) versus the proposed increment  $z$ ; for a logistic regression example with real data.

Carlo approximation. MAMBA selects a step-size equal to  $3.24 \times 10^{-4}$  and  $1.36 \times 10^{-3}$  for v-SGLD and v-SGBD respectively. In general, the chosen step-sizes match the largest scale of the coordinates but are too large for the first one. We note that SGBD outperforms SGLD in particular for the first coordinate where SGLD remarkably inflates its variance.

Figure 14 reports the density estimates for all coordinates of the posterior samples from a third experiment where we selected different step-sizes for each coordinated. In particular, we are interested in the case where a diagonal preconditioner is applied and we set  $\sigma_{1,j} = 0.1 \times sd(\pi_j)$  and  $\sigma_{2,j} = 0.2 \times sd(\pi_j)$ , for  $j = 1, \dots, 4$ , where  $sd(\pi_j)$  denotes the standard deviation of the posterior distribution for  $\theta_j$ . The aim of this experiment is to study the performance of the algorithm with a correct tuning of the step-size across the coordinates. However, we note that this is in general not easily to do in practice as it requires to have access to posterior quantities which are in general unknown a priori and usually estimated with MCMC. In this scenario, clearly both algorithms perform better than in the previous one and sample accurately with small step-sizes (dotted lines in Figure 14). With a larger step-size (dashed lines in Figure 14), the samplers moderately inflate the variance of the marginals distribution with SGBD performing slightly better.

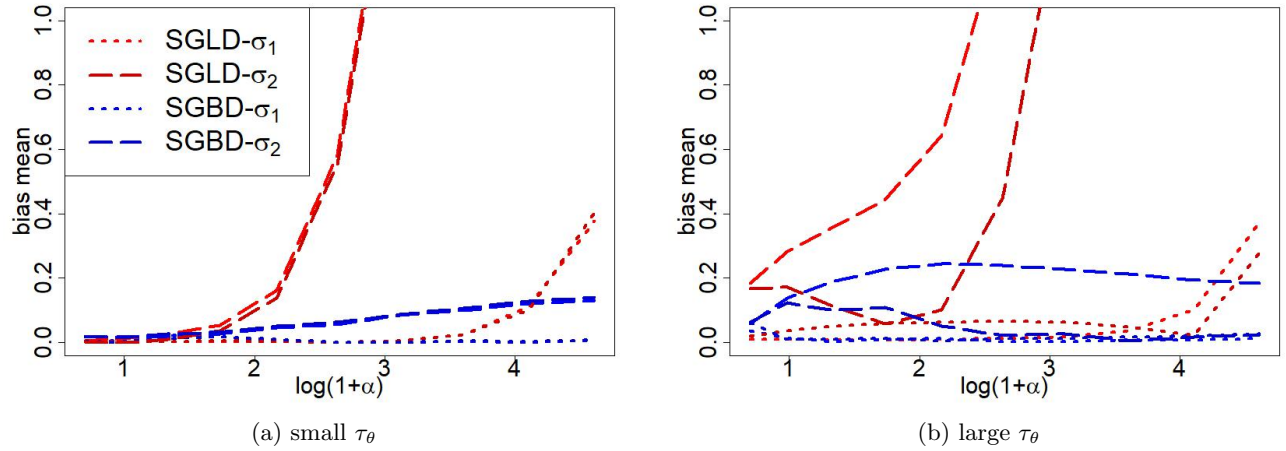


Figure 9. Toy Example: Skew-Normal target with isotropic Gaussian Noise in gradient estimates. Shape parameter (in log-scale) versus relative bias of mean with noise standard deviation  $\tau_\theta = 1 \times sd(\pi_\alpha)$  (left) and  $\tau_\theta = 10 \times sd(\pi_\alpha)$  (right). Red refers to Langevin-based schemes and blue to Barker-based schemes. Lighter lines refer to vanilla implementations of the algorithm, darker lines to their corrected variants. Dotted (dashed resp.) lines are produced with  $\sigma_1 = 0.1 \times sd(\pi_\alpha)$  ( $\sigma_2 = 0.5 \times sd(\pi_\alpha)$  resp.).

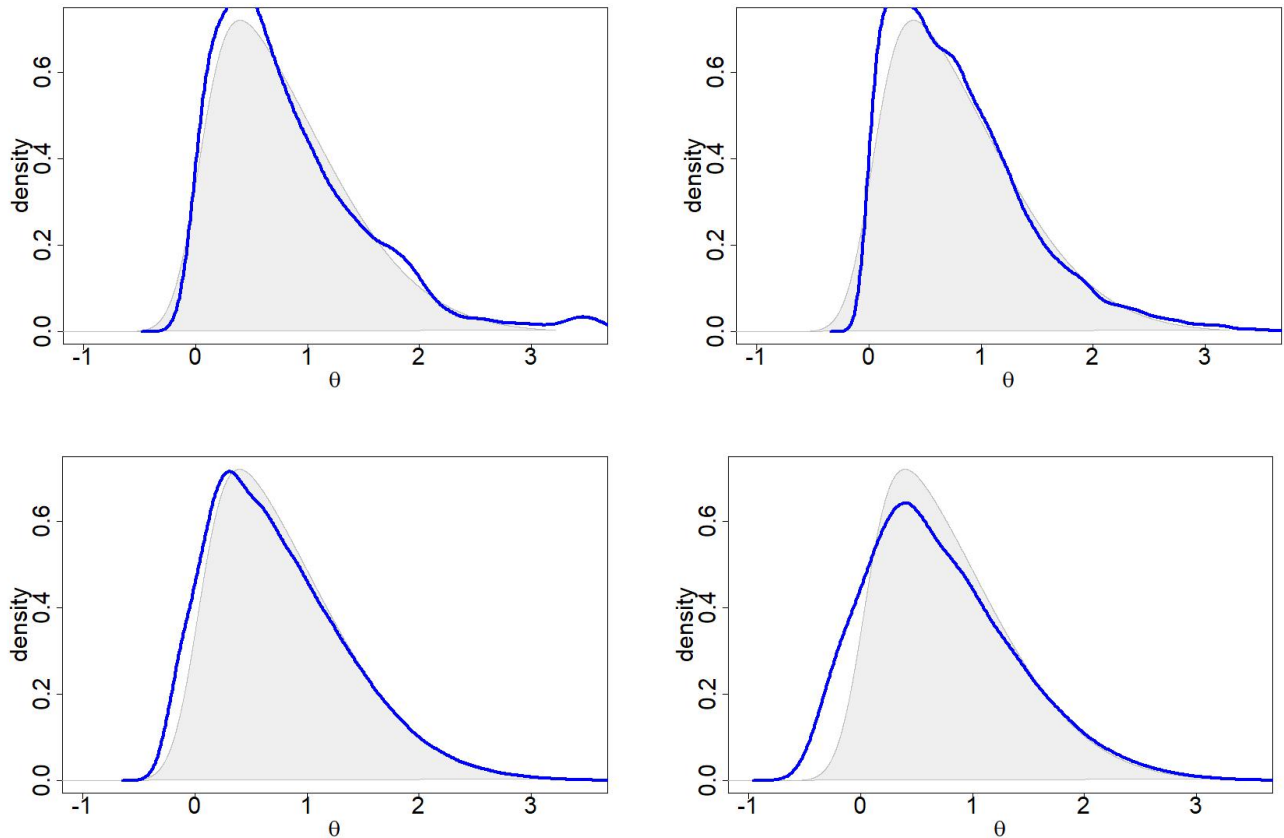


Figure 10. Toy Example: Skew-Normal target with isotropic Gaussian Noise in gradient estimates. Density estimates of the samples obtained via v-SGBD with the step-size respectively equal to  $0.05 \times sd(\pi_\alpha)$ ,  $0.1 \times sd(\pi_\alpha)$ ,  $0.5 \times sd(\pi_\alpha)$ , and  $0.75 \times sd(\pi_\alpha)$ , for  $\alpha = 20$ .

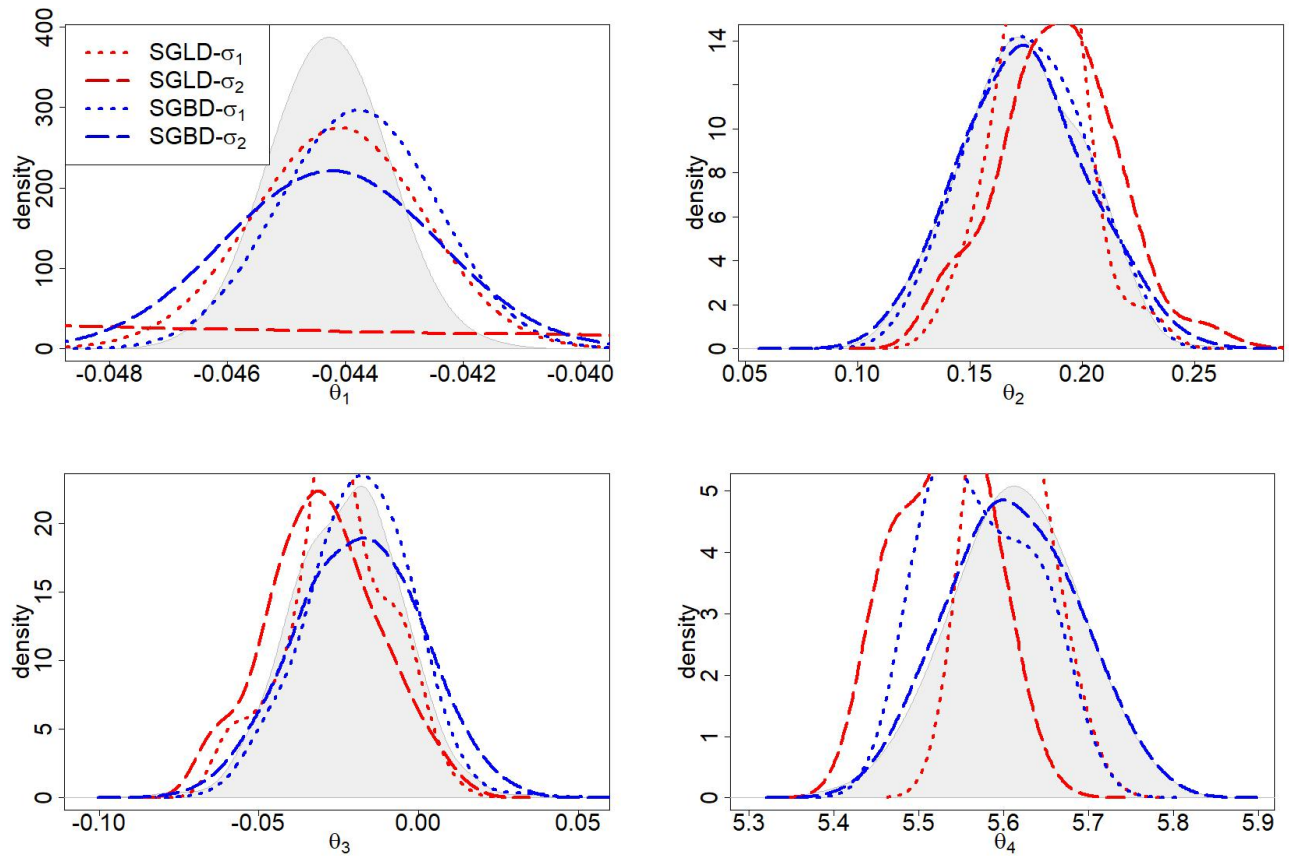
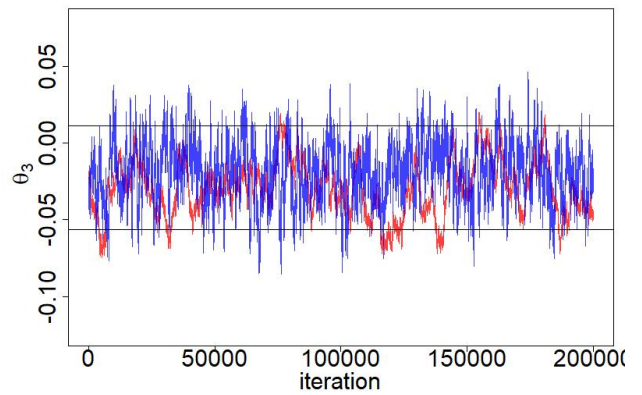
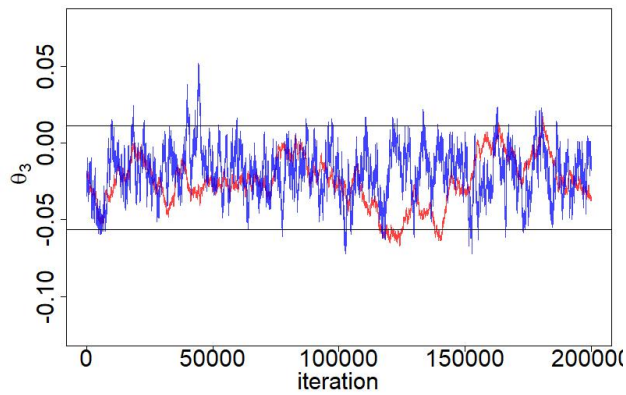
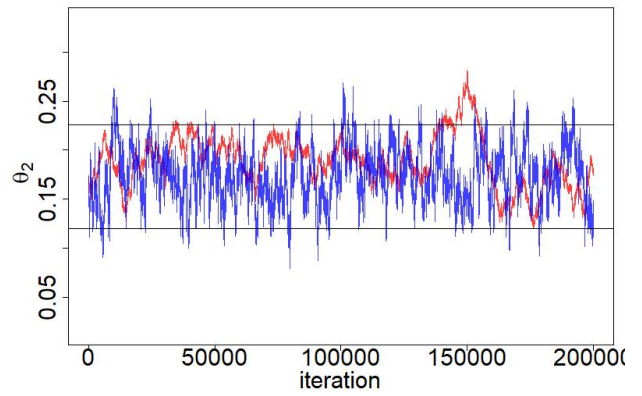
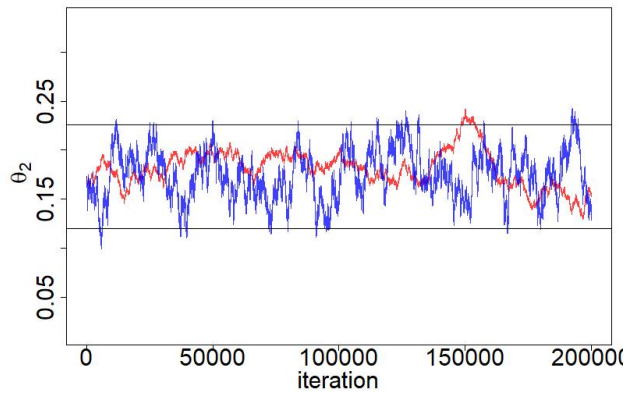
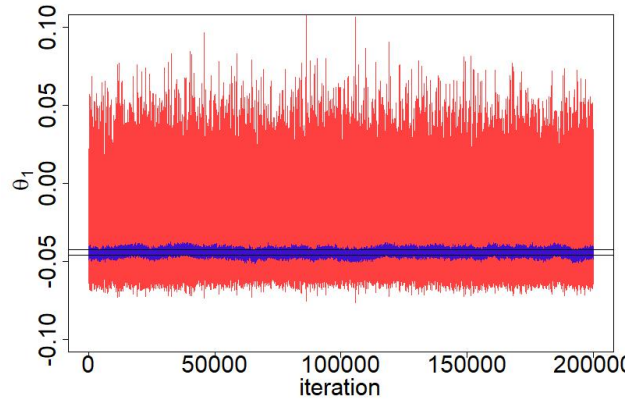
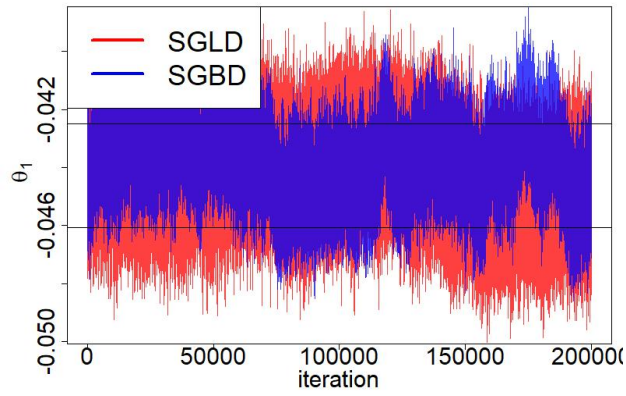


Figure 11. Logistic Regression with Scale Heterogeneity on the Sepsis dataset. Blue (red resp.) lines represent the density estimates of v-SGBD (v-SGLD) samples. Dotted (dashed resp.) lines are produced using a smaller (larger resp.) step size. Grey areas represent the estimate of the marginal posterior densities obtained with STAN.



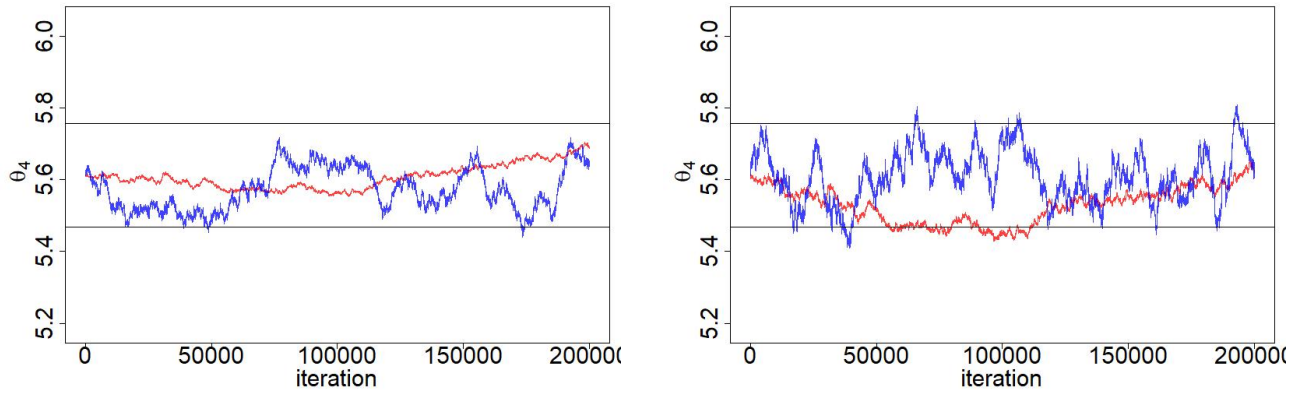


Figure 12. Logistic Regression with Scale Heterogeneity on the Sepsis dataset. Traceplots of all the coordinates with two step-size configurations: small  $\sigma$  (left), large  $\sigma$  (right). Blue refers to v-SGBD and red to v-SGLD. Black horizontal lines represent the interval around the posterior mean with a two standard deviations width.

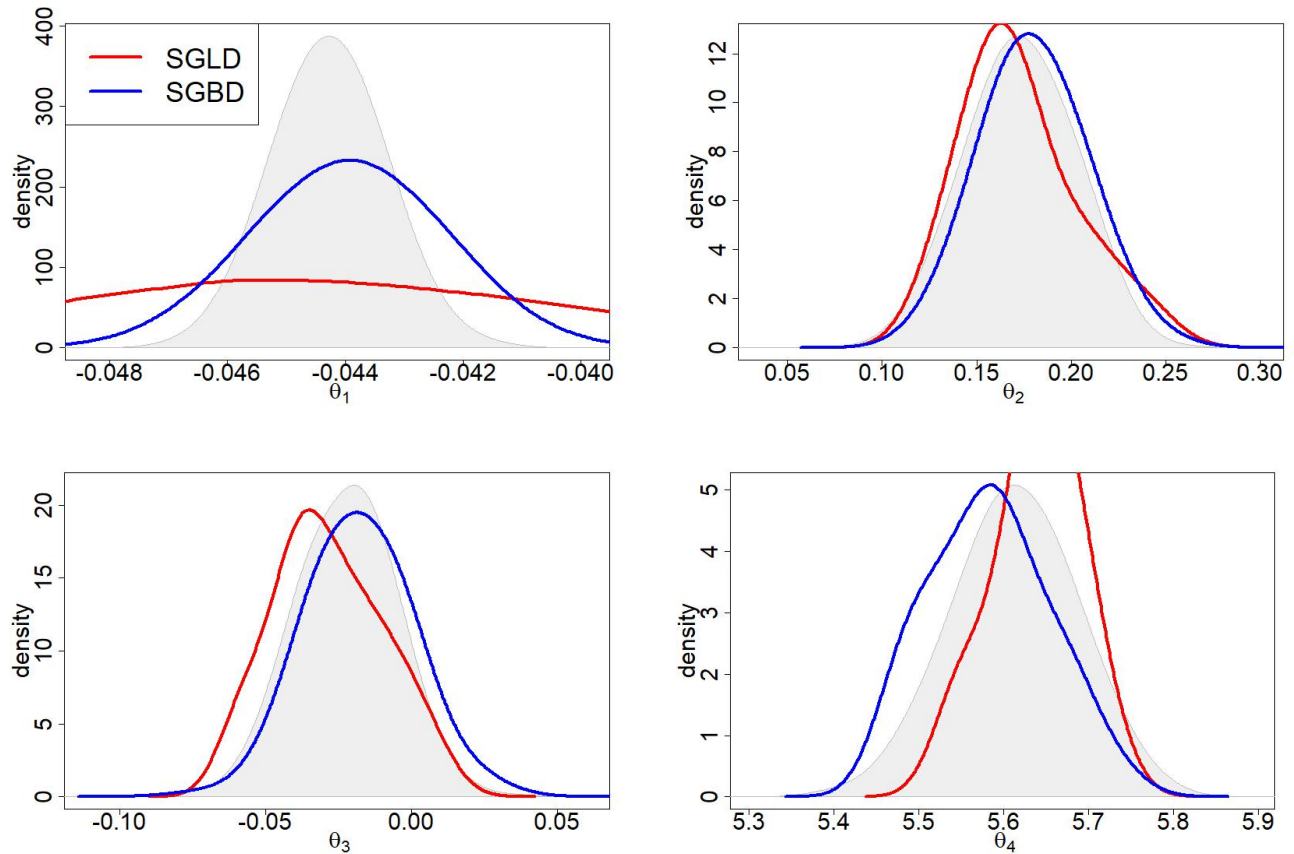


Figure 13. Logistic Regression with Scale Heterogeneity on the Sepsis dataset. Blue (red resp.) lines represent the density estimates of v-SGBD (v-SGLD) samples tuning the step-size via MAMBA. Grey areas represent the estimate of the marginal posterior densities obtained with STAN.

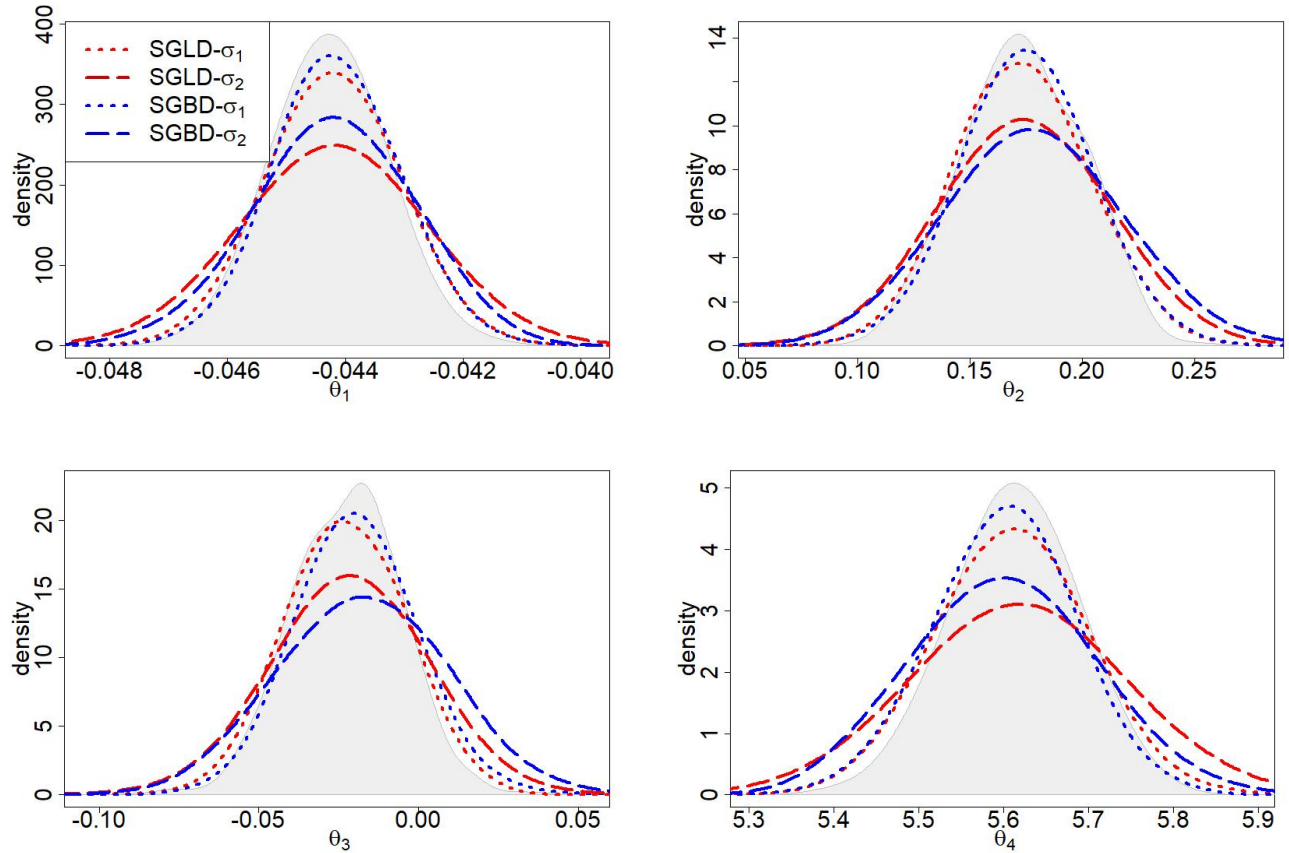


Figure 14. Logistic Regression with Scale Heterogeneity on the Sepsis dataset. Blue (red resp.) lines represent the density estimates of v-SGBD (v-SGLD) samples using different step-sizes for each coordinate. Dotted (dashed resp.) lines are produced with  $\sigma_{1,j} = 0.1 \times sd(\pi_j)$  ( $\sigma_{2,j} = 0.2 \times sd(\pi_j)$  resp.) for  $j = 1, \dots, 4$ . Grey areas represent the estimate of the marginal posterior densities obtained with STAN.



#### D.4 Binary Regression With High-Dimensional Predictors

This section studies the performance of SGBD on ill-conditioned high dimensional logistic regression task using model (14). We apply the model to the Arrhythmia dataset from the UCI repository. The dataset contains 452 instances and 279 covariates, from which we retain the first 100. A random 80 – 20% train-test split is applied to the dataset, and we run the samplers for  $T = 10^5$ , discarding the first half as burn-in, iteration using a mini-batch of  $n = 34$ .

We are interested in how hyperparameter tuning affects the sampling accuracy of the algorithm. In particular, we study the trade-off between mixing and sampling accuracy, since increasing the step size of the algorithms produces better mixing but less accurate chains, as no MH step is used. Figure 15 shows how sampling accuracy decreases as mixing increases when the step sizes vary. Mixing is measured with the median effective sample size (ESS) across the parameters and sampling accuracy with the mean standardized  $1^{st}$  and  $2^{nd}$  order bias, which are defined as follows:

$$\begin{aligned} \text{Bias}(\mathbb{E}[\theta_j | \{y_i\}_{i=1}^n]) &= \frac{|\bar{\theta}_j - \mathbb{E}[\theta_j | \{y_i\}_{i=1}^n]|}{(\mathbb{V}[\theta_j | \{y_i\}_{i=1}^n])^{1/2}} \\ \text{Bias}(\mathbb{V}[\theta_j | \{y_i\}_{i=1}^n]) &= \frac{|\bar{\tau}_{\theta_j}^2 - \mathbb{V}[\theta_j | \{y_i\}_{i=1}^n]|}{(\mathbb{V}[\theta_j | \{y_i\}_{i=1}^n])^{1/2}} \end{aligned} \quad j = 1, \dots, d, \quad (40)$$

where  $\bar{\theta}_j = \frac{\sum_{t=1}^T \theta_j^{(t)}}{T}$  and  $\bar{\tau}_{\theta_j}^2 = \frac{\sum_{t=1}^T (\theta_j^{(t)} - \bar{\theta}_j)^2}{T-1}$ . The results in Figure 15 suggests that SGLD is more accurate for

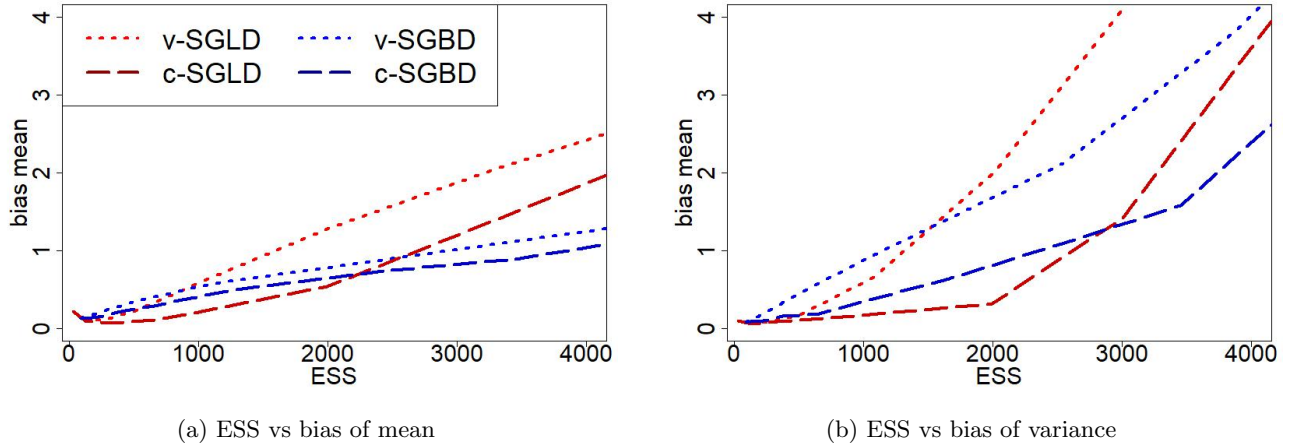


Figure 15. Logistic Regression Mixing-Accuracy trade off on the Arrhythmia dataset. Median ESS versus mean bias of mean (left) and mean bias of variance (right). Red refers to SGLD and blue to the SGBD. Lighter dotted lines refer to vanilla implementations of the algorithm, darker dashed lines to their corrected variants.

small step-sizes and low mixing, while SGBD appears to be more robust to hyperparameter tuning. In particular, it enjoys a more favourable mixing-accuracy trade off when the step size is chosen increasingly large.

Figure 16a reports the predictive performance in terms of log-loss on the held-out test set, when hyperparameters are chosen such that the median ESS of the samples is roughly equal to 1000 and are set to  $\sigma = 0.25, 0.22, 0.07$  for v-SGBD, c-SGBD and e-SGBD and  $\sigma = 0.14, 0.11, 0.09$  for v-SGLD, c-SGLD and e-SGLD. The log-loss at iteration  $t$  is defined as

$$l(t) = -\frac{1}{|T|} \sum_{i \in T} y_i \log(\hat{p}(\mathbf{x}_i, \theta)^{(t)}) + (1 - y_i) \log(1 - \hat{p}(\mathbf{x}_i, \theta)^{(t)}) \quad (41)$$

where  $T$  is the test set,  $|T|$  denotes its size, and  $\hat{p}(\mathbf{x}_i, \theta)^{(t)} = \frac{1}{t} \sum_{k=1}^t (1 + e^{-\mathbf{x}_i^\top \theta^{(k)}})^{-1}$  is the ergodic average of the estimate of probability of  $Y_i = 1$  given the predictors and samples for the parameter  $\theta$ . SGBD outperforms SGLD in terms of predictive accuracy.

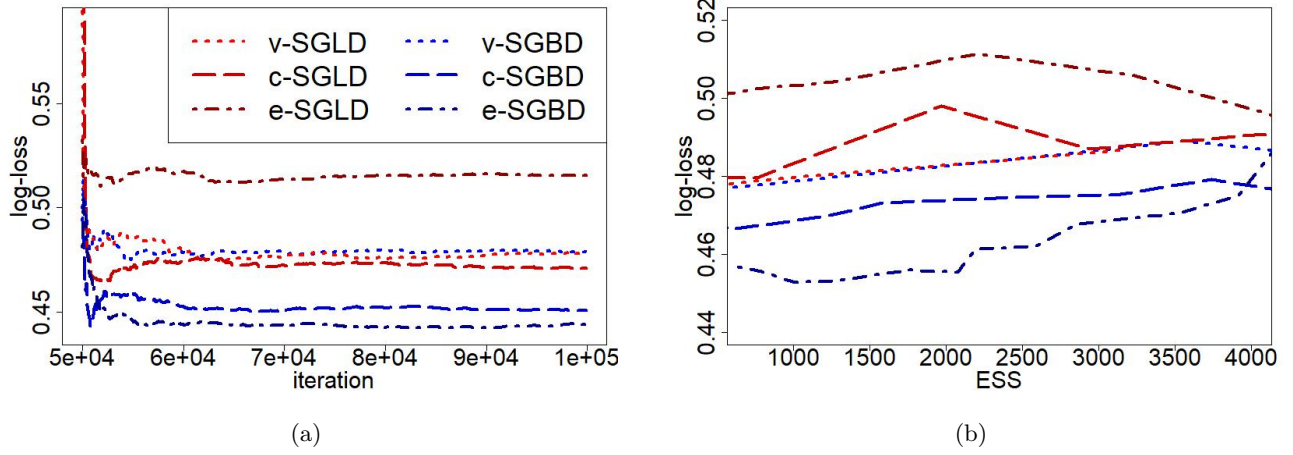


Figure 16. Logistic Regression Predictive Accuracy on the Arrhythmia dataset. Iteration versus log-loss of MCMC estimates for a given configuration of step-size (left) and Median ESS versus log-loss using all samples after the burn-in (right). Red refers to SGLD and blue to the SGBD. For both algorithms, the vanilla (lighter dotted lines), corrected (medium scale dashed lines) and extreme (darker dotted-dashed lines) versions are displayed.

Figure 16b reports how the log-loss using all the samples from each chain varies for each configuration of hyperparameter. In general, SGBD achieves a better predictive accuracy than SGLD with different step-sizes configurations.

#### D.5 Additional details for the numerics in Sections 4.3 and 4.4

This section reports additional details about the experiments in Section 4.3 and 4.4 of the paper.

In Section 4.3, we run the algorithms with the following value for the hyperparameter  $\sigma = 0.022$  and  $\sigma = 0.005$  for v-SGLD and e-SGLD and  $\sigma = 0.011$  and  $\sigma = 0.0105$  for v-SGBD and e-SGBD. The rMSE reported in Figure 4 is computed after clipping the ratings predicted values at 1 and 5. In particular, the sample rMSE (s-rMSE) at iteration  $t$  is computed as

$$\text{s-rMSE}(t) = \sqrt{\frac{1}{|T|} \sum_{i,j:(i,j) \in T} (R_{ij} - \hat{R}_{ij}^{(t)})^2}, \quad \hat{R}_{ij}^{(t)} = \begin{cases} 1 & \text{if } \tilde{R}_{ij}^{(t)} < 1, \\ 5 & \text{if } \tilde{R}_{ij}^{(t)} > 5, \\ \tilde{R}_{ij}^{(t)} & \text{otherwise} \end{cases}, \quad (42)$$

where  $\tilde{R}_{ij}^{(t)} = \mathbf{U}_i^{(t)\top} \mathbf{V}_j^{(t)}$ ,  $T$  is the test-set and  $|T|$  denotes its size. The rMSE of the ergodic average of the predictions (e-rMSE) at iteration  $t$  is computed as

$$\text{e-rMSE}(t) = \sqrt{\frac{1}{|T|} \sum_{i,j:(i,j) \in T} (R_{ij} - \bar{R}_{ij}^{(t)})^2}, \quad (43)$$

where  $\bar{R}_{ij}^{(t)} = \frac{1}{t} \sum_{k=1}^t \hat{R}_{ij}^{(k)}$  is the ergodic average of the predictions for  $R_{ij}$ . Similar results were obtained without clipping the predictions.

In Figure 5 of Section 4.4, the sample mean log-likelihood (s-L) at iteration  $t$  is computed as

$$\text{s-L}(t) = \frac{1}{|T|} \sum_{i \in T} \log(p(x_i | W^{(t)})) \quad (44)$$

and the mean log-likelihood of the ergodic average of  $W$  (e-L) at iteration  $t$  is given by

$$\text{e-L}(t) = \frac{1}{|T|} \sum_{i \in T} \log(p(x_i | \bar{W}^{(t)})) \quad (45)$$

where  $\bar{W}^{(t)} = \frac{1}{t} \sum_{k=1}^t W^{(k)}$ ,  $T$  is the test-set and  $|T|$  denotes its size. We chose the following step-sizes optimizing predictive performance of the ergodic average of the samples of  $W$  over a grid. obtaining  $\sigma = 0.0110$  for v-SGBD,  $\sigma = 0.0084$  for c-SGBD and e-SGBD,  $\sigma = 0.0070$  for v-SGLD and c-SGLD and  $\sigma = 0.0063$  for e-SGLD.