Journal of Machine Learning Research 20 (2019) 1-31 Submitted 8/17; Revised 5/19; Published 7/19

Hamiltonian Monte Carlo with Energy Conserving Subsampling

Khue-Dung Dang DOANKHUEDUNG.DANG@UNSW.EDU.AU
UNSW Business School, School of Economics, University of New South Wales.
ARC Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS)

Matias Quiroz QUIROZ.MATIAS@GMAIL.COM
School of Mathematical and Physical Sciences, University of Technology Sydney.
ARC Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS)

Robert Kohn R.KOHN@UNSW.EDU.AU
UNSW Business School, School of Economics, University of New South Wales.
ARC Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS)

Minh-Ngoc Tran MINH-NGOC.TRAN@SYDNEY.EDU.AU
Discipline of Business Analytics, University of Sydney.
ARC Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS)

Mattias Villani MATTIAS.VILLANI@LIU.SE
Division of Statistics and Machine Learning, Linköping University.
Department of Statistics, Stockholm University.
ARC Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS)

Editor: Ryan Adams

Abstract

Hamiltonian Monte Carlo (HMC) samples efficiently from high-dimensional posterior distributions with proposed parameter draws obtained by iterating on a discretized version of the Hamiltonian dynamics. The iterations make HMC computationally costly, especially in problems with large data sets, since it is necessary to compute posterior densities and their derivatives with respect to the parameters. Naively computing the Hamiltonian dynamics on a subset of the data causes HMC to lose its key ability to generate distant parameter proposals with high acceptance probability. The key insight in our article is that efficient subsampling HMC for the parameters is possible if both the dynamics and the acceptance probability are computed from the same data subsample in each complete HMC iteration. We show that this is possible to do in a principled way in a HMC-within-Gibbs framework where the subsample is updated using a pseudo marginal MH step and the parameters are then updated using an HMC step, based on the current subsample. We show that our subsampling methods are fast and compare favorably to two popular sampling algorithms that use gradient estimates from data subsampling. We also explore the current limitations of subsampling HMC algorithms by varying the quality of the variance reducing control variates used in the estimators of the posterior density and its gradients.

Keywords: Bayesian inference, Big Data, Markov chain Monte Carlo, Estimated likelihood, Stochastic gradient Hamiltonian Monte Carlo, Stochastic Gradient Langevin Dynamics

1. Introduction

Bayesian inference relies on computing expectations with respect to the posterior density of the model parameters given the data. The functional form of the posterior density often does not correspond to a known density and hence obtaining independent samples to compute the expectation by Monte Carlo integration is difficult, especially when the dimension of the model parameter is moderate to large. Markov Chain Monte Carlo (MCMC) is a generic sampling algorithm that produces correlated draws from the posterior density.

Metropolis-Hastings (MH) (Metropolis et al., 1953; Hastings, 1970) is arguably the most popular MCMC algorithm. Its most common implementation uses a random walk proposal, in which a new sample is proposed based on the current state of the Markov chain. While Random walk MH is easy to implement, it explores the posterior very slowly in high-dimensional problems and gives highly correlated draws and imprecise estimators of posterior integrals.

Hamiltonian Monte Carlo (HMC) (Duane et al., 1987) can produce distant proposals while maintaining a high acceptance probability (Neal, 2011; Betancourt, 2017). HMC augments the target posterior by adding fictitious momentum variables and carries out the sampling on an extended target density. The extended target is proportional to the exponential of a Hamiltonian function that describes the total energy of the system, which is the sum of the potential energy (negative log posterior) and the kinetic energy (negative log density of the momentum variables). The Hamiltonian dynamics describes how the total energy evolves through time. One particularly interesting feature of the Hamiltonian is that it conserves energy as time evolves, a property that is approximately maintained even when the dynamics is approximated in discrete time. Hence, a MH proposal obtained by simulating the dynamics has approximately the same value of the extended target density as that of the current draw, resulting in a high acceptance probability, even when the proposed draw is far from the current draw. This typically avoids the inherently slow exploration of the parameter space evident in random walk proposals (Betancourt, 2017). HMC simulates the evolution of Hamiltonian dynamics for a given period of time. Up to a point, the longer the integration time the more effectively the dynamics explore the posterior distribution, while small integration times approach diffusive Langevin methods (Roberts and Rosenthal, 1998; Roberts and Stramer, 2002). In practice, the simulation of the dynamics is implemented with a numerical integrator. The more accurate the integrator, the larger the step size and the fewer total steps needed to simulate the dynamic for a chosen time. Smaller step sizes typically require larger integration times to achieve the same efficient exploration of the posterior distribution. In either case, HMC requires computing the gradient of the log-posterior in each step when simulating the dynamics, and in practice a large number of steps may be performed. The extra computations are often worthwhile as HMC greatly improves the sampling efficiency of the generated samples compared to a proposal which does not use gradient information. Even with the improved performance of HMC, however, the cost can be too prohibitive for the limited computational resources available in a given application. Consequently, some algorithms use subsampling of the data to reduce the cost of computing the posterior density and its gradients. Unfortunately, naive subsampling methods are often problematic and in particular they compromise many of the

features that give HMC its scalability (Betancourt, 2015). In this article we introduce a subsampling HMC scheme that can tackle very large data sets and maintain the scalability.

Our article speeds up computation by using subsets of the data to compute both the dynamics and the subsequent MH correction performed when deciding to accept a proposal. More precisely, we propose a HMC-within-Gibbs algorithm that alternates i) sampling small subsets of data using a pseudo-marginal step and ii) sampling parameters using the HMC dynamics and the MH correction based on the current subset of the data. We will here focus on HMC algorithms where the Hamiltonian dynamics are used to generate a proposal which is subsequently accepted or rejected using a MH step, which we refer to as HMC. Extensions to other HMC algorithms that use the entire trajectory generated by the dynamics (Hoffman and Gelman, 2014; Betancourt, 2017) are interesting future research direction discussed in Section 7.

We propose two different subsampling versions of the HMC algorithm. In the first *perturbed* approach we use a slightly biased likelihood estimator and show that the algorithm targets a perturbed posterior which gets close to the true posterior density as the subsample size and the number of observations of the data set become large; see Section 4.2 for details. The second approach uses an unbiased but possibly negative likelihood estimator which allows us to obtain simulation consistent estimators of the posterior expectation of any function of the parameters. However, this approach is harder to implement efficiently than the perturbed approach and is typically slower computationally.

We compare our algorithms to Stochastic Gradient Langevin Dynamics (Welling and Teh, 2011, SGLD) and Stochastic Gradient Hamiltonian Monte Carlo (Chen et al., 2014, SG-HMC), two of the most popular subsampling algorithms that use gradient information in machine learning. To make the comparison more challenging we implement both methods with control variates for improved performance (Baker et al., 2017). We demonstrate that our algorithms compare favorably to SGLD and SG-HMC. It is by now well known that all proposed subsampling MCMC and HMC algorithms need accurate control variates to lower the variance of the estimated posterior and gradients, and we explore the robustness of the algorithms when the control variates are degraded.

The paper is organized as follows. Section 2 reviews previous research and the methods we compare against. Section 3 presents our methodology using a general likelihood estimator and argues that it circumvents the incompatibility of data subsampling and HMC raised in recent literature. Sections 4 and 5 present two algorithms based on two specific likelihood estimators. Finally, Section 6 demonstrates the usefulness of the methods on two large data sets and compares with alternative approaches. This section also explores the limitation of subsampling approaches by experimenting with successively degrading the quality of the control variates used for variance reduction. Section 7 concludes and discusses future research.

2. Related work

2.1. High-dimensional MCMC for large data sets

There has recently been a surge of interest in developing posterior simulation algorithms that scale with respect to both the number of observations n and the number of parameters d. Since simulation methods have the ambitious goal of exploring all regions in parame-

ter space with sizable probability mass, they naturally require many more iterations than posterior optimization algorithms. Posterior optimization is computationally attractive for big data, but does not quantify the posterior uncertainty, which is often a central task in science. Although there exist optimization-based methods that aim to approximate the entire posterior distribution, e.g. variational Bayes (Blei et al., 2017), Laplace approximations (Bernardo and Smith, 2001, Chapter 5) or integrated nested Laplace approximations (Rue et al., 2009), in practice it is nearly impossible to know how they perform without comparing the results to a posterior simulation method. It is thus important to develop posterior simulation methods that:

i) remain computationally efficient when n is large and

ii) explore the posterior distribution efficiently when d is large.

Two distinct approaches exist to resolve i). The first is to use parallel computing by dividing the n data observations into K parts, performing independent posterior simulation on each of the K subposteriors and subsequently merge the draws to represent the full data posterior. See, for example, Scott et al. (2013); Neiswanger et al. (2014); Wang and Dunson (2014); Minsker et al. (2014); Nemeth and Sherlock (2018). The second approach, which is the focus of our article, is to work with subsamples of m observations to estimate the full data posterior (Maclaurin and Adams, 2014; Korattikara et al., 2014; Bardenet et al., 2014, 2017; Maire et al., 2018; Bierkens et al., 2019; Quiroz et al., 2019, 2018b,a; Gunawan et al., 2018) or its gradient (Welling and Teh, 2011; Chen et al., 2014; Ma et al., 2015; Shang et al., 2015; Baker et al., 2017). The rest of this section reviews samplers which use gradient information about the posterior density.

The primary problem confronted in ii) is how to generate proposals which maintain a high acceptance probability and are also distant enough to avoid a highly persistent Markov chain for the model parameter. A useful approach is to simulate a discretized Langevin diffusion (Roberts and Rosenthal, 1998; Nemeth et al., 2016) or, more generally, Hamilton's equations (Duane et al., 1987) and use the simulated draw as a proposal in the MH algorithm to correct for the bias introduced by the discretization (Neal, 2011). HMC provides a solution to ii) (Neal, 2011; Betancourt, 2017), but when combined with i), the algorithm becomes computationally intractable since simulating the Hamiltonian dynamics requires a large number of costly gradient evaluations for every proposed parameter value.

2.2. Subsampling HMC algorithms and related approaches

A computationally attractive way to accelerate HMC is to use a fixed subsample of the data to unbiasedly estimate the computationally costly gradients in each step of the discretized Hamiltonian trajectory, and skip the MH correction to avoid evaluating the posterior on the full data. Betancourt (2015) demonstrates that this simple strategy produces highly biased trajectories, where the bias depends upon the quality of the gradient estimator. Moreover, Betancourt (2015) shows that attempts to average out the bias by renewing the subsample in each step of the trajectory still perform poorly; see also the *naive stochastic gradient HMC* in Chen et al. (2014). Betancourt (2015) illustrates that adding a MH correction step based on the full data to fully correct for the biased trajectories leads to a rapid deterioration of the acceptance probability of HMC as d increases, and concludes that there is a fundamental

incompatibility of HMC and data subsampling. As a remedy to the poor performance by the naive stochastic gradient HMC, Chen et al. (2014) propose adding a friction term to the dynamics to correct for the bias. For the rest of our article, we refer to the method using the friction term in the dynamics as Stochastic Gradient Hamiltonian Monte Carlo (SG-HMC). Chen et al. (2014) omit the MH correction step and the resulting bias in the posterior depends on the quality of the discretization of the continuous dynamics. Hence, in order to traverse the parameter space effectively, the integrator potentially needs a large number of steps to compensate for the small step size used when discretizing the dynamics, which may be very costly.

A different approach, but related in the sense that it uses an estimated gradient, is Stochastic Gradient Langevin Dynamics (Welling and Teh, 2011, SGLD). SGLD combines Stochastic Gradient Optimization (Robbins and Monro, 1951, SGO) and Langevin dynamics (Roberts and Rosenthal, 1998), by allowing the initial iterates to resemble a SGO and gradually traverse to Langevin Dynamics so as to not collapse to the posterior mode. SGLD avoids a costly MH correction by arguing that it is not needed because the discretization step size of the Langevin dynamics is decreased as a function of the iterates (Welling and Teh, 2011). However, this decreases the rate of convergence of estimators based on its output to $R^{-1/3}$, where R is the number of samples from the posterior (Teh et al., 2016), as opposed to $R^{-1/2}$ for MCMC (Roberts and Rosenthal, 2004), and in particular HMC. Practical implementations of SGLD use a sequence of step sizes that does not decrease to zero in the limit, and Vollmer et al. (2016) show that the posterior approximated by SGLD can then be quite far from the true posterior; see also Brosse et al. (2018). Bardenet et al. (2017) also demonstrate that SGLD can be accurate for the posterior mode, but gives a poor approximation of the full posterior distribution on a toy example with $d = 2$ and highly redundant data, i.e. superfluous amounts of data in relation to the complexity of the model. Recently, Dubey et al. (2016) improve SGLD using control variates, see also Baker et al. (2017) who, in addition, use control variates in the SG-HMC algorithm proposed in Chen et al. (2014). We implement both SGLD and SG-HMC with highly efficient control variates when comparing them to our method. We also implement the methods without control variates as it has been shown that sometimes variance reduction may be detrimental (Chatterji et al., 2018).

All the problems discussed above with subsampling in HMC and related algorithms stem from the fact that subsampling disconnects the Hamiltonian from its own dynamics. This disconnect causes HMC proposals to lose their energy conserving property and their attractive ability to sample efficiently in high dimensions. The next section presents a new energy conserving approach to subsampling in HMC that keeps the connection intact by estimating both the Hamiltonian and its corresponding dynamics from the same subsample. By updating the subsample in a separate pseudo-marginal step we make sure that the HMC algorithm still targets the posterior based on all data. Put differently, our new approach creates a Hamiltonian system with corresponding dynamics for a given subset of data. This allows for the scalability of HMC to be maintained for each subsample, as is demonstrated in Section 6.7.

3. Energy conserving subsampling in HMC

This section presents our new approach to subsampling in HMC and discusses why our approach avoids the pitfalls described in Betancourt (2015). In order to not distract from the main ideas, we first present our approach for a general unbiased and almost surely positive likelihood estimator from data subsampling. Sections 4 and 5 then present practical algorithms based on likelihood estimators proposed in Quiroz et al. (2019) and Quiroz et al. (2018b), respectively.

3.1. Pseudo-marginal MCMC

Denote the model parameter vector by $\theta \in \Theta \subset \mathbb{R}^d$, where $\mathbb{R}^d$ is the space of d-dimensional real vectors, and let $\pi(\theta)$ be its posterior density given a data set y with n observations. We first briefly describe pseudo-marginal MCMC (Andrieu and Roberts, 2009), which serves as inspiration for one building block in our subsampling HMC approach. Pseudo-marginal algorithms target the augmented posterior

$$\overline{\pi}_m(\theta, u) \propto \widehat{L}_m(\theta) p_\Theta(\theta) p_U(u), \tag{1}$$

where $\widehat{L}_m(\theta)$ is an unbiased and non-negative estimator of the likelihood $L(\theta)$ based on m auxiliary variables, u, with density $p_U(u)$. In the particular application to subsampling, $u \in \{1, \ldots, n\}^m$, contains the indices for the data observations used in estimating the likelihood and m denotes the subsample size, see Section 4.1 for details. Note that $\widehat{L}_m$ and any quantity included in its definition depend on n, but this is suppressed in the notation for conciseness. We can now design an MCMC chain to sample θ and u jointly from (1) and, since $\widehat{L}_m(\theta)$ is unbiased, the θ iterates are samples from $\pi(\theta)$.

The choice of m is crucial for pseudo-marginal methods. An m that is too small results in a noisy likelihood estimator and the Markov chain may get stuck due to severely overestimating the likelihood at the current draw, subsequently rejecting nearly all proposals. Conversely, taking m too large wastes useful computational resources. A natural aim is to choose an m that minimizes the Computational Time (CT) needed to generate the equivalent of a single independent draw from the posterior, with

$$\text{CT} \coloneqq \text{IF} \times \text{Total number of density and gradient evaluations}, \tag{2}$$

where the Inefficiency Factor (IF) is proportional to the asymptotic variance when estimating a posterior functional based on the MCMC output, and is interpreted as the number of samples needed to obtain the equivalent of a single independent sample. The second term is proportional to the computing time for a single draw, a measure that is independent of the implementation. Starting with Pitt et al. (2012), there is a large literature showing that CT is minimized by an m that targets a variance of the log of the likelihood estimator around 1 (Doucet et al., 2015; Sherlock et al., 2015; Tran et al., 2017; Deligiannidis et al., 2018; Schmon et al., 2018). Recent developments in pseudo-marginal methods induce dependence in the auxiliary variables u over the MCMC iterations such that the likelihood estimates over the iterations become dependent (Deligiannidis et al., 2018). This makes it possible to tolerate a substantially larger variance of the likelihood estimator, and hence smaller subsamples in our context. We follow Tran et al. (2017) and induce the dependence over

the iterations by partioning the u's into blocks and only update a subset of the blocks in each iteration. The optimal subsample size m is then obtained by targeting a certain value for the conditional variance of the likelihood estimator for a given induced correlation ρ (Tran et al., 2017).

3.2. An energy conserving HMC-within-Gibbs framework

Following the standard HMC algorithm, our subsampling HMC algorithm introduces a fictitious continuous momentum vector $\vec{p} \in \mathcal{P} \subset \mathbb{R}^d$ of the same dimension as the continuous parameter vector θ. The extended target in (1) is then further augmented by $\vec{p}$ to

$$\overline{\pi}_m(\theta, \vec{p}, u) \propto \exp\left(-\widehat{\mathcal{H}}(\theta, \vec{p})\right) p_U(u), \quad \widehat{\mathcal{H}}(\theta, \vec{p}) = \widehat{\mathcal{U}}(\theta) + \mathcal{K}(\vec{p}) \tag{3}$$

with

$$\widehat{\mathcal{U}}(\theta) = -\log \widehat{L}_m(\theta) - \log p_\Theta(\theta) \quad \text{and } \mathcal{K}(\vec{p}) = \frac{1}{2}\vec{p}\,' M^{-1} \vec{p}, \tag{4}$$

where M is a symmetric positive-definite matrix. In (3) we assume that the Hamiltonian $\widehat{\mathcal{H}}$ is separable. We propose a HMC-within-Gibbs method to sample from (3), alternating sampling from

1. $u|\theta, \vec{p}, y$ - Pseudo-marginal MH update (Section 3.3)
2. $\theta, \vec{p}\,|u, y$ - HMC update given u from Step 1 (Section 3.4).

This scheme has (3) as its invariant distribution. Integrating out the momentum variables yields $\overline{\pi}_m(\theta, u)$ in (1) and, further integrating out u, yields $\pi(\theta)$ if the likelihood estimator $\widehat{L}_m(\theta)$ is unbiased. Lindsten and Doucet (2016) propose the related pseudo-marginal HMC sampler, in which a momentum vector is also introduced for the auxiliary variables u. That scheme, however, is not applicable here as the pseudo-marginal variables we employ are discrete and not amenable to Hamiltonian dynamics themselves.

The next subsections describe in detail the two updates of our algorithm and explain why our approach does not compromise the Hamiltonian flow.

3.3. Updating the data subset

Given $\theta^{(j-1)}, \vec{p}^{\,(j-1)}$ and $u^{(j-1)}$, at iteration j we propose $u' \sim p_U(u)$ and set $u^{(j)} = u'$ with probability

$$\alpha_u = \min\left\{1, \frac{\widehat{L}_m(\theta^{(j-1)}; u')}{\widehat{L}_m(\theta^{(j-1)}; u^{(j-1)})}\right\}, \tag{5}$$

where the notation emphasizes that the estimators are based on different data subsets. If the proposal is rejected we set $u^{(j)} = u^{(j-1)}$.

Since u' is proposed independently of $u^{(j-1)}$, the log of the ratio in (5) can be highly variable, possibly getting the sampler stuck when the numerator is significantly overestimated. To prevent this, we implement the block update of u for data subsampling in Quiroz et al. (2019, 2018b) with G blocks, which gives a correlation ρ of roughly $1 - 1/G$ between the $\log \widehat{L}_m$ at the current and proposed draws (Tran et al., 2017; Quiroz et al., 2018b). Setting $G = 100$, gives $\rho \approx 0.99$, which helps the chain to mix well.

3.4. Updating the parameters

Given $u^{(j)}$, we use Hamilton's equations

$$\frac{d\theta_l}{dt} = \frac{\partial \widehat{\mathcal{H}}(\theta, \vec{p})}{\partial \vec{p}_l}, \quad \frac{d\vec{p}_l}{dt} = -\frac{\partial \widehat{\mathcal{H}}(\theta, \vec{p})}{\partial \theta_l}, \quad l = 1, \ldots, d, \tag{6}$$

to propose θ and $\vec{p}$. Note that this trajectory follows the Hamiltonian flow for $\widehat{\mathcal{H}}$ viewed as a function of θ and $\vec{p}$ for a given data subset selected by $u^{(j)}$, since $u^{(j)}$ is fixed through time t. We obtain the proposal as in standard HMC, using a leapfrog integrator with integration time ϵL, but with $\widehat{\mathcal{H}}$ in place of $\mathcal{H}$. Specifically, at iteration j, given the data subset $u^{(j)}$, if the leapfrog integrator starts at $(\theta^{(j-1)}, \vec{p}_0)$ with $\vec{p}_0 \sim \mathcal{K}(\vec{p})$ and ends at $(\theta_L, -\vec{p}_L)$, we let $(\theta^{(j)}, \vec{p}^{\,(j)}) = (\theta_L, -\vec{p}_L)$ with probability

$$\alpha_{\theta,\vec{p}} = \min\left\{1, \exp\left(-\widehat{\mathcal{H}}(\theta_L, -\vec{p}_L) + \widehat{\mathcal{H}}(\theta^{(j-1)}, \vec{p}_0)\right)\right\}, \tag{7}$$

with $\widehat{\mathcal{H}}$ in (3). If $(\theta_L, -\vec{p}_L)$ is rejected, we set $(\theta^{(j)}, \vec{p}^{\,(j)}) = (\theta^{(j-1)}, \vec{p}_0)$. In practice, it is unnecessary to store the sampled momentum.

Using the terminology in Betancourt (2015), we can think of the dynamics in (6) as generating a trajectory following a *modified level set* ($\widehat{\mathcal{H}}$), as opposed to the *exact level set* obtained using dynamics that do not subsample the data ($\mathcal{H}$). A key property of our framework is that the *same* estimate $\widehat{\mathcal{H}}$ is used in generating the discretized leapfrog trajectory as in the acceptance probability in (7). The connection between the modified level set $\widehat{\mathcal{H}}$ and its dynamics is kept intact, and thus the original energy conserving property of HMC remains, even for distant proposals. We therefore name our algorithm Hamiltonian Monte Carlo with Energy Conserving Subsampling (HMC-ECS). Note that energy conservation is only possible because of the pseudo-marginal mechanism where we update the subsample in each Gibbs iteration, thereby guaranteeing that our samples are from the target posterior based on all the data.

Betancourt (2015) illustrates the problems with using $\widehat{\mathcal{H}}$ for the dynamics, but $\mathcal{H}$ in the acceptance probability. Given a sensible step length ϵ, discretizing the Hamiltonian with a symplectic integrator introduces an error of $O(\epsilon^2)$ (Neal, 2011) relative to the modified level set and hence the discretization error is very small. Betancourt (2015) notes that the modified level set and the discretized trajectory based on it might be very far from the exact level set, resulting in low acceptance probabilities no matter how small ϵ is. SG-HMC (Chen et al., 2014) deliberately circumvents the disconnect problem by generating proposals from the trajectories based on a modified Hamiltonian, but skip the rejection step. The disadvantage of SG-HMC is therefore that the bias in the targeted posterior now grows with the step length ϵ. Keeping ϵ small makes SG-HMC very computationally demanding since a very large number of leapfrog steps are needed for distant proposals. In contrast, the dynamics of HMC-ECS target the subsampled Hamiltonian, and so maintain a high acceptance probability even for a large ϵ. The bias introduced by the subsampling is then confined to the pseudo-marginal step, which is chosen so that the bias is very small as our theoretical analysis below shows.

Algorithm 1 shows one iteration of our proposed HMC-ECS algorithm based on the leapfrog integrator using the estimated likelihood $\widehat{L}_m(\theta)$. The next two sections consider previously proposed likelihood estimators and show how we use them in HMC-ECS.

Algorithm 1: One iteration of HMC-ECS.

Input: Current position $u^{(j-1)}$, $\theta^{(j-1)}$, stepsize ϵ and integrating time ϵL

Propose $u' \sim p_U(u)$

Set $u^{(j)} \leftarrow u'$ with probability

$$\alpha_u = \min\left\{1, \frac{\widehat{L}_m(\theta^{(j-1)}; u')}{\widehat{L}_m(\theta^{(j-1)}; u^{(j-1)})}\right\},$$

else $u^{(j)} \leftarrow u^{(j-1)}$

Given $u^{(j)}$:

$\vec{p}_0 \sim \mathcal{K}(\vec{p})$; $\theta_0 \leftarrow \theta^{(j-1)}$; $\widehat{\mathcal{H}}_0 \leftarrow \widehat{\mathcal{H}}(\theta^{(j-1)}, \vec{p}_0)$

$\vec{p}_0 \leftarrow \vec{p}_0 - \frac{\epsilon}{2}\nabla_\theta \widehat{\mathcal{U}}(\theta_0)$

for $l = 1$ **to** L **do**

  $\theta_l \leftarrow \theta_{l-1} + \epsilon M^{-1}\vec{p}_{l-1}$

  if $i < L$ **then** $\vec{p}_l \leftarrow \vec{p}_{l-1} - \epsilon\nabla_\theta \widehat{\mathcal{U}}(\theta_l)$;

  else $\vec{p}_L \leftarrow \vec{p}_{L-1} - \frac{\epsilon}{2}\nabla_\theta \widehat{\mathcal{U}}(\theta_l)$;

end

$\vec{p}_L \leftarrow -\vec{p}_L$

$\widehat{\mathcal{H}}_L \leftarrow \widehat{\mathcal{H}}(\theta_L, \vec{p}_L)$

Set $(\theta^{(j)}, \vec{p}^{\,(j)}) \leftarrow (\theta_L, \vec{p}_L)$ with probability

$$\alpha_{\theta,\vec{p}} = \min\left\{1, \exp\left(-\widehat{\mathcal{H}}(\theta_L, -\vec{p}_L) + \widehat{\mathcal{H}}(\theta^{(j-1)}, \vec{p}_0)\right)\right\},$$

else $(\theta^{(j)}, \vec{p}^{\,(j)}) \leftarrow (\theta^{(j-1)}, \vec{p}_0)$

Output: $u^{(j)}$, $\theta^{(j)}$

4. Perturbed HMC-ECS

4.1. Efficient estimators of the log-likelihood

Quiroz et al. (2019) propose sampling m observations with replacement and estimate an additive log-likelihood

$$\ell(\theta) = \log L(\theta) = \sum_{k=1}^{n} \ell_k(\theta), \quad \ell_k(\theta) = \log p(y_k|\theta)$$

by the unbiased difference estimator

$$\widehat{\ell}_m(\theta) = \sum_{k=1}^{n} q_k(\theta) + \widehat{d}_m(\theta), \tag{8}$$

where

$$\widehat{d}_m(\theta) = \frac{1}{m}\sum_{i=1}^{m} \frac{\ell_{u_i}(\theta) - q_{u_i}(\theta)}{\omega_{u_i}(\theta)}, \quad u_i \in \{1,\ldots,n\} \text{ iid with } \Pr(u_i = k) = \omega_k, \tag{9}$$

and $q_k(\theta)$ are control variates. We continue to suppress dependence on n in the notation for many quantities introduced in this section. If the $q_k(\theta)$ approximate the $\ell_k(\theta)$ reasonably well, then we obtain an efficient estimator by taking $\omega_k = 1/n$ for all k. Quiroz et al. (2019) estimate $\sigma^2(\theta) = \mathrm{V}\left[\widehat{\ell}_m(\theta)\right]$ by

$$\widehat{\sigma}^2_m(\theta) = \frac{n^2}{m^2}\sum_{i=1}^{m} \left(d_{u_i}(\theta) - \overline{d}_u(\theta)\right)^2, \quad \text{with } d_{u_i}(\theta) = \ell_{u_i}(\theta) - q_{u_i}(\theta) \tag{10}$$

where $\overline{d}_u$ denotes the mean of the d_{u_i} in the sample $u = (u_1,\ldots,u_m)$.

To obtain efficient control variates, Quiroz et al. (2019) follow Bardenet et al. (2017) and let $q_k(\theta)$ be a second order Taylor approximation around a fixed central value $\theta^\star$,

$$q_k(\theta) = \ell_k(\theta^\star) + \nabla_\theta \ell_k(\theta^\star)^\top(\theta - \theta^\star) + \frac{1}{2}(\theta - \theta^\star)^\top H_k(\theta^\star)(\theta - \theta^\star), \quad H_k(\theta^\star) \coloneqq \nabla_\theta \nabla_\theta^\top \ell_k(\theta^\star). \tag{11}$$

After processing the full data once before the MCMC to compute simple summary statistics, $\sum_{k=1}^{n} q_k(\theta)$ can be computed in $O(1)$ time (Bardenet et al., 2017).

4.2. Efficient estimators of the likelihood and its gradient

Quiroz et al. (2019) use the likelihood estimator

$$\widehat{L}_m(\theta) = \exp\left(\widehat{\ell}_m(\theta) - \frac{1}{2}\widehat{\sigma}^2_m(\theta)\right) \tag{12}$$

first proposed by Ceperley and Dewing (1999) and Nicholls et al. (2012). The motivation for this estimator is that it is unbiased for the likelihood if i) $\widehat{\ell}_m(\theta) \sim \mathcal{N}(\ell(\theta), \sigma^2(\theta))$ (justified by the central limit theorem) and ii) $\widehat{\sigma}^2_m(\theta)$ in (12) is replaced by the population quantity

$\sigma^2(\theta)$. However, $\sigma^2(\theta)$ is not available in practice and the estimator in (12) is biased. This bias makes the MCMC algorithm in Quiroz et al. (2019) target a slightly perturbed posterior $\overline{\pi}_m(\theta)$. Assuming that the expansion point $\theta^\star$ in the control variates is the posterior mode based on all the data, Quiroz et al. (2019) prove that

$$\int_\Theta |\overline{\pi}_m(\theta) - \pi(\theta)| d\theta = O\left(\frac{1}{nm^2}\right) \text{ and } |\mathrm{E}_{\overline{\pi}_m}[h(\theta)] - \mathrm{E}_\pi[h(\theta)]| = O\left(\frac{1}{nm^2}\right), \tag{13}$$

where $\overline{\pi}_m(\theta) = \mathrm{E}_{p_U}\left[\widehat{L}_m(\theta)\right]$ is the perturbed marginal for θ when using the likelihood estimator in (12). These results carry over to our Hamiltonian approach straightforwardly as we obtain the augmented target in Quiroz et al. (2019) after integrating out the momentum in (3) and using (12). Hence, the θ iterates from HMC-ECS converge to a perturbed posterior which may get arbitrarily close to the true posterior by increasing the subsample size m, or by increasing n and letting $m = O(n^\nu)$ for some $\nu > 0$. For example, if $\nu = 1/2$, then the above orders are $O(1/n^2)$ with respect to n. However, this extremely rapidly vanishing perturbation is usually not practically attainable since the result in (13) assumes that $\theta^\star$ is the posterior mode based on *all* data. Corollary 1 in Quiroz et al. (2019) proves rates under the more realistic assumption that $\theta^\star$ is the posterior mode based on a fixed subset of $\tilde{n} \ll n$ observations. If, for example, $\tilde{n} = O(\sqrt{n})$ then the rates in (13) become $O(1/\sqrt{n})$. Importantly, the optimal subsample size in this case becomes $m = O(\sqrt{n})$, which shows that HMC-ECS scales well with the size of the data. See Quiroz et al. (2019) for suggestions on how to get closer to the rates in (13) in a computationally tractable way.

In addition to estimating the likelihood and the log-likelihood, our Hamiltonian approach also needs to estimate a gradient. It is straightforward to modify (8) to instead provide an unbiased estimator of the gradient of the log-likelihood. With $\omega_k = 1/n$ and $\nabla_\theta q_k(\theta) = \nabla_\theta \ell_k(\theta^\star) + H_k(\theta^\star)(\theta - \theta^\star)$,

$$\nabla_\theta \widehat{\ell}_m(\theta) = A(\theta^\star) + B(\theta^\star)(\theta - \theta^\star) + \frac{n}{m}\sum_{i=1}^{m}\left(\nabla_\theta \ell_{u_i}(\theta) - \nabla_\theta q_{u_i}(\theta)\right), \tag{14}$$

where

$$A(\theta^\star) \coloneqq \sum_{k=1}^{n} \nabla_\theta \ell_k(\theta^\star) \in \mathbb{R}^d \quad \text{and } B(\theta^\star) \coloneqq \sum_{k=1}^{n} H_k(\theta^\star) \in \mathbb{R}^{d\times d}$$

are obtained at the cost of computing over the full data set once before the MCMC since $\theta^\star$ is fixed. It is also straightforward to compute $\nabla_\theta \widehat{\sigma}^2_m(\theta)$ using (10) with this choice of control variate.

It is important to note that the perturbation in the targeted posterior in perturbed HMC-ECS is independent of the step length in the leapfrog iterations, ϵ. Hence, we can generate distant proposals from a small number of leapfrog steps without increasing the posterior bias.

5. Signed HMC-ECS

We now present an alternative HMC-ECS algorithm based on the Block-Poisson estimator in Quiroz et al. (2018b). This algorithm gives simulation consistent estimates of expectations with respect to the true posterior density without any perturbation.

5.1. The block-Poisson estimator

Quiroz et al. (2018b) propose the block-Poisson estimator, formed by sampling $\mathcal{X}_l \sim \text{Pois}(1)$ for $l = 1, \dots \lambda$, and computing $\widehat{d}_m^{(h,l)}$, $h = 1, \dots, \mathcal{X}_l$, using (9) based on a mini-batch sample size m, and then estimate the likelihood by

$$\widehat{L}_m(\theta) = \exp\left(\sum_{k=1}^{n} q_k(\theta)\right) \prod_{l=1}^{\lambda} \xi_l, \qquad \xi_l = \exp\left(\frac{a+\lambda}{\lambda}\right) \prod_{h=1}^{\mathcal{X}_l} \left(\frac{\widehat{d}_m^{(h,l)}(\theta) - a}{\lambda}\right), \tag{15}$$

where λ is a positive integer, a is a real number and $\xi_l = \exp\left((a+\lambda)/\lambda\right)$ if $\mathcal{X}_l = 0$. Pois(1) denotes the Poisson distribution with mean 1. Note that the total subsample size $m\lambda\mathcal{X}_l$ is random with mean $m\lambda$ in a given MCMC iteration.

Since $\widehat{L}_m(\theta)$ is unbiased for $L(\theta)$, defining the augmented density as in (3) gives

$$\int_U \int_{\mathcal{P}} \overline{\pi}_m(\theta, \vec{p}, u) d\vec{p} du = \pi(\theta).$$

Hence, if (3) is a proper density using (15), we obtain samples from the desired marginal for θ. However, (3) is a proper density only if $\tau := \Pr\left(\widehat{L}_m(\theta) \geq 0\right) = 1$, which requires that a in (15) is a lower bound of $\widehat{d}_m^{(h,l)}$ (Jacob and Thiery, 2015) which results in a prohibitively costly estimator (Quiroz et al., 2018b). Instead, we follow Quiroz et al. (2018b) who use the approach of Lyne et al. (2015) for exact inference on an expectation of an arbitrary function $\psi(\theta)$ with respect to $\pi(\theta)$. For our Hamiltonian approach, this entails defining a proper augmented density,

$$\widetilde{\pi}_m(\theta, \vec{p}, u) \propto \left|\widehat{L}_m(\theta)\right| p_\Theta(\theta) p_U(u) \exp\left(-\mathcal{K}(\vec{p})\right), \tag{16}$$

and writing

$$\text{E}_\pi[\psi] = \frac{\int_\Theta \psi(\theta) L(\theta) p_\Theta(\theta) d\theta}{\int_\Theta L(\theta) p_\Theta(\theta) d\theta} = \frac{\int_{\mathcal{U}} \int_{\mathcal{P}} \int_\Theta \psi(\theta) S(\theta, u) \widetilde{\pi}(\theta, \vec{p}, u) d\theta d\vec{p} du}{\int_{\mathcal{U}} \int_{\mathcal{P}} \int_\Theta S(\theta, u) \widetilde{\pi}(\theta, \vec{p}, u) d\theta d\vec{p} du} = \frac{\text{E}_{\widetilde{\pi}}[\psi S]}{\text{E}_{\widetilde{\pi}}[S]}, \tag{17}$$

where $S(\theta, u) = \text{sign}\left(\widehat{L}_m(\theta)\right)$ and $\text{sign}(\cdot) = 1$ if $\widehat{L}_m(\theta) \geq 0$ and $\text{sign}(\cdot) = -1$ otherwise. Equation (17) suggests running the HMC-ECS sampler outlined in Section 3.2 on the target (16), and then estimating (17) by

$$\widehat{I}_R = \frac{\sum_{j=1}^{R} \psi(\theta^{(j)}) s^{(j)}}{\sum_{j=1}^{R} s^{(j)}},$$

where $s^{(j)}$ is the sign of the estimate at the jth iteration. We follow Quiroz et al. (2018b) and use the term *signed PM* for any pseudo-marginal algorithm that uses the technique in Lyne et al. (2015) where a pseudo-marginal sampler is run on the absolute value of the estimated posterior and subsequently sign-corrected by importance sampling. Similarly, we call the algorithm described in this section *signed HMC-ECS*.

The block-Poisson estimator in (15) has more tuning parameters than the estimator in (12). Quiroz et al. (2018b) extend the optimal tuning approach in Pitt et al. (2012) to the

signed pseudo-marginal algorithm with the block-Poisson estimator. The Computational Time (CT) measure in (2) now becomes

$$\text{CT} \coloneqq \frac{\text{IF}}{(2\tau-1)^2} \times \text{Total number of density and gradient evaluations}, \tag{18}$$

where $\tau := \Pr\left(\widehat{L}_m(\theta) \geq 0\right)$. Quiroz et al. (2018b) derive analytical expressions for both $\text{V}[\log|\widehat{L}_m|]$ and $\Pr(\widehat{L}_m \geq 0)$ needed to optimize CT. The fact that Quiroz et al. (2018b) take $\Pr(\widehat{L}_m \geq 0)$ into account when tuning the algorithm avoids the instability from changing signs in signed PMMH. Quiroz et al. (2018b) also consider optimal tuning when correlating the estimators of the log of the likelihood at the current and proposed values of the MCMC. This correlation, ρ, is achieved by only updating u for a subset of the products in (15) in each iteration, keeping the others fixed. Quiroz et al. (2018b) show that if u is updated in κ products, then $\rho \approx 1 - \kappa/\lambda$.

6. Applications

6.1. Model

We consider the logistic regression

$$p(y_k|x_k,\theta) = \left(\frac{1}{1+\exp(-x_k^\top\theta)}\right)^{y_k}\left(\frac{1}{1+\exp(x_k^\top\theta)}\right)^{1-y_k}, \quad \text{with } p_\Theta(\theta) = \mathcal{N}(\theta|0,\lambda_\theta^{-2}I),$$

where λ_θ is a global shrinkage factor which we treat as constant for simplicity. We estimate the model on two large data sets described below.

6.2. Competing algorithms and performance measure

We compare the performance of both the perturbed and the signed HMC-ECS algorithms against SGLD and SG-HMC. All subsampling methods use the same control variates based on a second order Taylor expansion for comparability; see Section 6.8 for experiments with control variates based on lower order Taylor expansions. The expansion point $\theta^\star$ is unique to each experiment and is discussed later.

Following Vollmer et al. (2016); Baker et al. (2017), we implement SGLD using a fixed small step size ϵ instead of decreasing it (which gives worse results). This gives the following dynamics after discretization

$$\theta_i = \theta_{i-1} - \frac{\epsilon}{2}\nabla_\theta\widehat{\mathcal{U}}(\theta_i) + \zeta_i, \quad i = 1,\dots,R,$$

where $\widehat{\mathcal{U}}(\theta) = -\nabla_\theta\widehat{\ell}_m(\theta) - \log p_\Theta(\theta)$ with $\nabla_\theta\widehat{\ell}_m(\theta)$ in (14) and $\zeta_i \sim \mathcal{N}(0,\epsilon I)$.

Following Chen et al. (2014), we implement SG-HMC using a discretized dynamics with momentum $\vec{p}$ with covariance matrix M of the form

$$\begin{aligned}\theta_l &= \theta_{l-1} + \epsilon M^{-1}\vec{p}_{l-1}\\ \vec{p}_l &= \vec{p}_{l-1} - \epsilon\nabla_\theta\widehat{\mathcal{U}}(\theta_l) - \epsilon C M^{-1}\vec{p}_{l-1} + \zeta_l, \quad l = 1,\dots,L,\end{aligned}$$

where $\widehat{\mathcal{U}}(\theta) = -\nabla_\theta \widehat{\ell}_m(\theta) - \log p_\Theta(\theta)$ with $\nabla_\theta \widehat{\ell}_m(\theta)$ in (14), $\zeta_l \sim \mathcal{N}(0, 2(C - \widehat{B})\epsilon)$. We set $\widehat{B} = 0$ (Chen et al., 2014) and $C = I$ (Ma et al., 2015).

The algorithms are compared with respect to CT as defined in (2) when the likelihood estimator is non-negative, where IF is computed using the CODA package in R (Plummer et al., 2006). For signed HMC-ECS, the CT is given by (18) and we estimate τ by the fraction of positive signs. Note that CT does not take into account that some of the algorithms can give a substantially biased estimate of the posterior, and we assess the bias separately.

The Relative Computational Time (RCT) between algorithm $\mathcal{A}_1$ and $\mathcal{A}_2$ is defined as

$$\text{RCT}_{\mathcal{A}_1,\mathcal{A}_2} = \frac{\text{CT}_{\mathcal{A}_2}}{\text{CT}_{\mathcal{A}_1}}. \tag{19}$$

6.3. Tuning and settings of our algorithms

We first outline how our algorithms are tuned. These settings are used for all the experiments unless stated otherwise.

The choice of the positive-definite mass matrix M in (4) is crucial for the performance of any HMC type algorithm: an M that closely resembles the covariance of the posterior facilitates sampling, especially when θ is highly correlated in the posterior (Neal, 2011; Betancourt, 2017). In logistic regression, we set $M = -\Sigma^{-1}(\theta^\star)$, where $\Sigma(\theta^\star)$ is the Hessian of the log posterior evaluated at some $\theta^\star$. We initialize $M = I$ and, during a burn-in period of 1,000 iterations, update M every 200 iterations based on the new $\theta^\star$. We use the same tuning of M in HMC-ECS when updating θ and $\vec{p}$ conditional on the data subsample u. However, since it is impractical to compute the Hessian of the conditional posterior at each iteration, we use the full data set when evaluating M (and include the cost in the CT), which performs well in practice although we stress that it is not optimal.

To select the step size ϵ in the leapfrog integrator, we use the dual averaging approach of Hoffman and Gelman (2014), which requires a predetermined trajectory length ϵL. We find that $\epsilon L = 1.2$ is sensible for our examples. The dual averaging algorithm uses this trajectory length and adaptively changes ϵ during the burn-in period in order to achieve a desired level of acceptance rate δ. We follow Hoffman and Gelman (2014) and set $\delta = 0.8$. The tuning for the logistic regression is relatively simple since Σ can be computed analytically, which is useful for both setting M and ϵ. More complex models are harder to tune, but this is unlikely to influence the comparisons between HMC-ECS and the other algorithms here, which is our primary concern. We stress that these tuning issues are inherent to HMC itself and not due to our subsampling approach. This strategy gives $\epsilon = 0.2$ and $L = 6$ for our algorithm which is the default in our experiments unless otherwise stated.

In HMC-ECS we generate subsamples with a correlation of $\rho = 0.99$. The subsample size m in the perturbed HMC-ECS approach is set according to the guidelines in Quiroz et al. (2019), see Section 4. In the signed HMC-ECS approach, we follow Quiroz et al. (2018b) who set the mini-batch size $m = 30$ and set λ optimally to minimize CT according to the formulas in Quiroz et al. (2018b). For our examples, $\lambda = 100$ and $\lambda = 200$, for the HIGGS and bankruptcy data sets, respectively.

6.4. Tuning and settings of the competing algorithms

We run full data HMC using the tuning strategy for ϵ and M outlined in Section 6.3 and use $M = \Sigma^{-1}(\theta^\star)$ with $\theta^\star$ as the posterior mean in SG-HMC. This favours SG-HMC over HMC-ECS, which needs to learn $\theta^\star$ as the algorithm progresses.

We are unaware of any tuning strategies for setting ϵ in SGLD and SG-HMC and therefore use trial and error. We find that the value of ϵ depends on which data set or algorithm was considered; see Sections 6.5 and 6.6. For SG-HMC we also need to set the number of steps L and we explore two choices. First, notice that HMC-ECS uses the trajectory length $\epsilon L = 1.2$. We thus set $L = 1.2/\epsilon$ so that SG-HMC can traverse the space as swiftly as HMC-ECS. We also compare with SG-HMC using a value of L which gives the same number of likelihood and gradient evaluations as HMC-ECS. For SGLD, we run the algorithm for R iterations that correspond to the number of gradient evaluations used post burn-in in HMC-ECS. For example, $R = 12{,}000$ iterations if $L = 6$ and HMC-ECS performs $2,000$ iterations.

Finally, we use the same m as HMC-ECS for SGLD and SG-HMC, as it is outside the scope of this paper to derive optimality results for those algorithms.

6.5. Results for the HIGGS data

Baldi et al. (2014) use the HIGGS data set, which contains 11 million observations, with a binary response *detected particle* predicted by 29 covariates. We use 10.5 millions observations for training and 500,000 for testing.

Unless stated otherwise, we start all algorithms at a $\theta^\star$ obtained as the posterior mean from running HMC on 1% of a randomly chosen subset of the data. This $\theta^\star$ is also used to initialize the control variates. We first run the algorithms using a full mass matrix M, chosen as explained above. The subsample size was set to $m = 1,300$ for all methods. For SG-HMC and SGLD, $\epsilon = 0.06$ and $\epsilon = 0.000001$ are used, respectively. The resulting L is 20 for SG-HMC.

Table 1 displays the CT for each algorithm compared to the perturbed HMC-ECS algorithm. The table shows the minimum, median and maximum RCT across all parameters. The best algorithm is perturbed HMC-ECS closely followed by signed HMC-ECS, both have RCTs that are roughly three times better than the best competitor SG-HMC. Although our metrics do not allow for direct comparison between biased (our approaches, SG-HMC and SGLD) methods and unbiased methods (HMC), we note that implementing HMC using the full data in this example is, for the perturbed and exact approaches respectively, 642.8 and 554.1 times more expensive in terms of posterior density and gradient evaluations.

RCT does not take the bias into account, however. Figure 1 displays kernel density estimates of the marginal posterior of four randomly selected parameters, and Figure 2 plots the posterior mean and variance for all the parameters from each algorithm against the true posterior mean and variance obtained from HMC based on the full data set. Figures 1 and 2 clearly show that both HMC-ECS algorithms and SG-HMC do a very good job in approximating the posterior, while SGLD gives biased estimates. We have also added the results for SG-HMC and SGLD without control variates as variance reduction is not always optimal for these algorithms (Chatterji et al., 2018). In this example, control variates

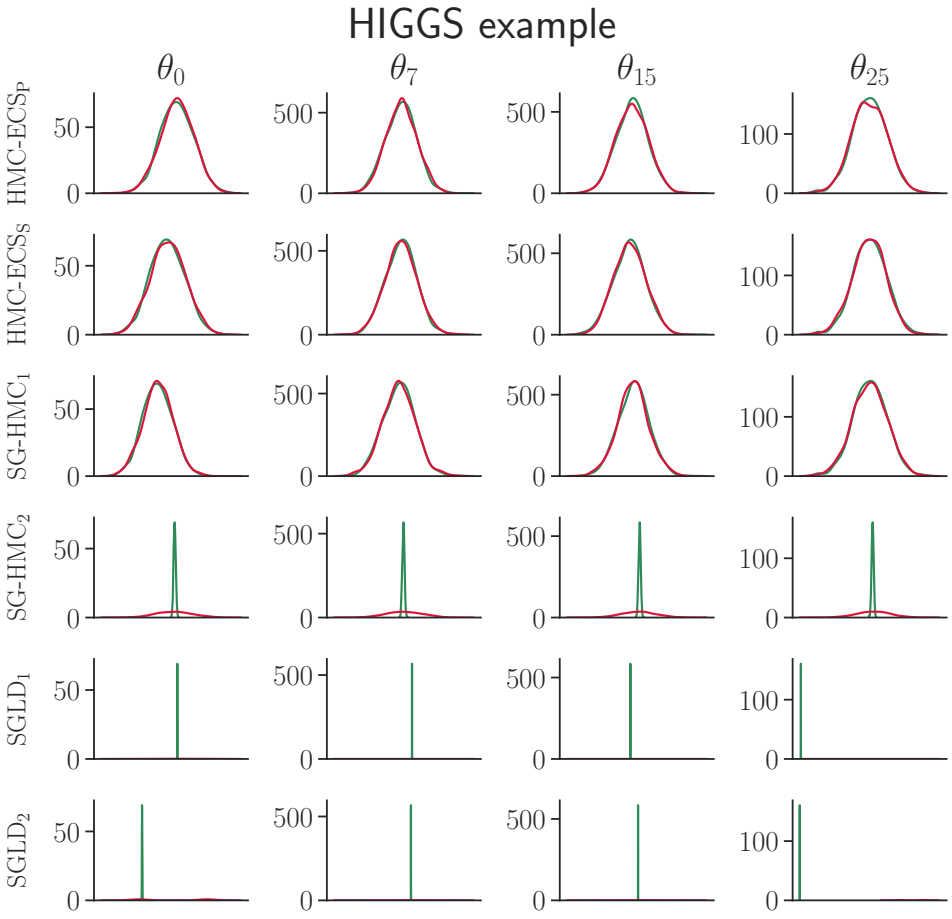

Figure 1: HIGGS data. The figure shows kernel density estimates of the posterior from the compared subsampling algorithms for four randomly selected parameters (green lines) and the corresponding posterior from HMC on the full data set (red lines). All algorithms use the negative inverse Hessian from all the data as mass matrix. HMC-ECS$_P$ and HMC-ECS$_S$ denote, respectively, the perturbed and signed HMC-ECS$_S$. For SGLD and SG-HMC, subscript 1 refers to the second order control variate and subscript 2 refers to the version without control variates.

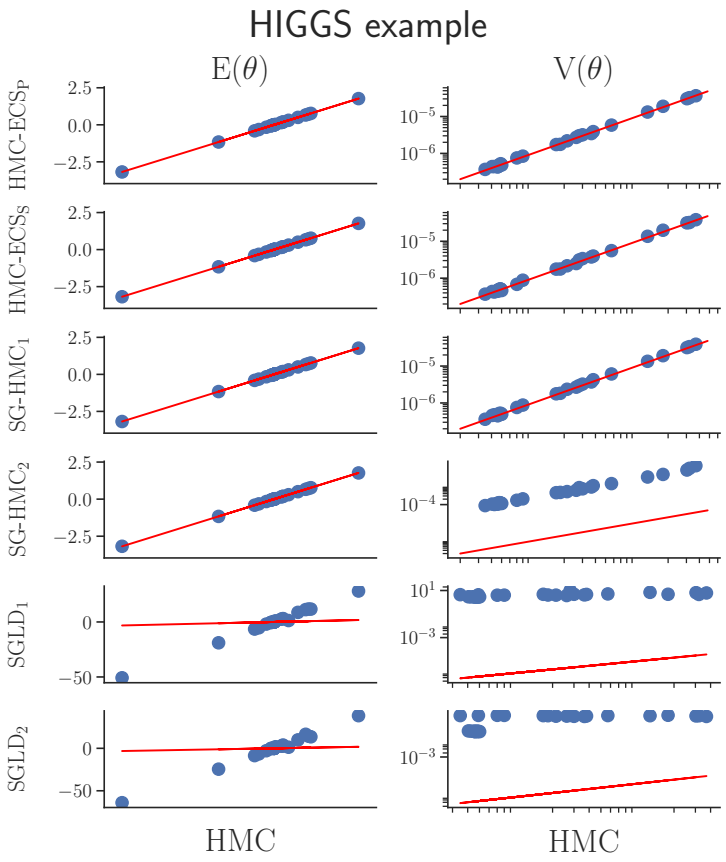

Figure 2: HIGGS data. The figure plots the estimated posterior mean and variance for all parameters from the subsampling algorithm against their true values obtained by HMC on the full data set. All algorithms use the negative inverse Hessian from all the data as mass matrix. HMC-ECS$_\text{P}$ and HMC-ECS$_\text{S}$ denote, respectively, the perturbed and signed HMC-ECS$_\text{S}$. For SGLD and SG-HMC, subscript 1 refers to the second order control variate and subscript 2 refers to the version without control variates.

RCT	Signed HMC-ECS	SG-HMC	SGLD
min	0.8	2.43	3.58
median	1.15	2.97	12.46
max	1.95	4.08	326.80

Table 1: HIGGS data. Relative computational time compared to perturbed HMC-ECS. The computational cost is (number of likelihood/gradient evaluations)×IF. For HMC-ECS and signed HMC-ECS the cost is computed for the entire run including training and warmup period. The RCT for the stochastic gradient methods are based on post-burnin iterations only.

	Perturbed HMC-ECS	Signed HMC-ECS
$\alpha_{\theta,\vec{p}}$	0.980	0.979
α_u	1	0.993
L	6	6
IF	2.185	2.192
ESS	927	922
$\widehat{\tau}$		1
$100(m/n)$	0.012	0.029

Table 2: HIGGS data. Summary of settings and efficiencies of HMC-ECS. The table shows the average acceptance probabilities (as a benchmark HMC has 0.980) in the post burn-in period for the two Gibbs steps, the number of steps L in the integrator used to obtain a predetermined trajectory length $\epsilon L = 1.2$, the average Inefficiency Factor (IF) (as a benchmark HMC has 2.084), the Effective Sample Size ESS $= R/\text{IF}$, the estimated probability of a positive likelihood estimator τ and the percentage of data used by each of the algorithms.

are indeed helpful, except for SGLD which does not provide an accurate approximation regardless of which control variate is used.

We conclude this example by demonstrating that HMC-ECS can safely be used for obtaining the predictive distribution. Figure 3 shows that the Receiver Operating Characteristic (ROC) curve for the $500,000$ test observations obtained with either of the two HMC-ECS algorithms are indistinguishable from the ROC curve obtained with HMC on the full data set.

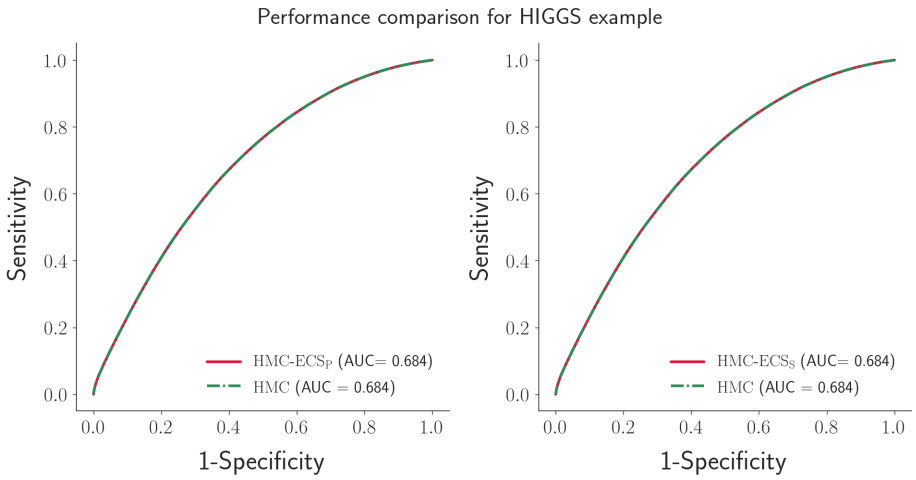

Figure 3: Prediction performance for the HIGGS data. The figure shows the Receiver Operating Characteristic (ROC) curves for the 500,000 test observations with HMC and HMC-ECS$_P$ (left panel, perturbed HMC-ECS) and HMC-ECS$_S$ (right panel, signed HMC-ECS).

6.6. Results for the Bankruptcy data

This data set contains annual observations on the bankruptcy status (binary y) of Swedish firms in the time period 1991-2008. We follow Giordani et al. (2014) and model the log odds of the firm failure probability as a non-linear function of six firm-specific financial ratios and two macroeconomic variables using an additive spline model. Giordani et al. (2014) estimate the model using frequentist methods. We use a Bayesian approach with an 81 dimensional posterior distribution (an intercept and 10 basis spline functions for each covariate) given the $n = 4{,}748{,}089$ firm-year observations.

Experimentation shows that $\epsilon = 0.02$ is a sensible choice for SG-HMC. Using the same trajectory length as in HMC-ECS gives $L = 60$ for SG-HMC. We also compare with SG-HMC using $L = 7$ for which SG-HMC has the same number of gradient evaluations as HMC-ECS has gradient and likelihood evaluations. For SGLD, $\epsilon = 0.00002$ is a sensible choice.

The subsample size for perturbed HMC-ECS was initially set to the optimal $m = 62{,}000$ following the guidelines in Quiroz et al. (2019) based on the initial value for $\theta^\star$. We then ran perturbed HMC-ECS for 100 iterations to obtain a better $\theta^\star$ and recalibrated to the now optimal $m = 1{,}000$ for this improved $\theta^\star$. This improved $\theta^\star$ is also used to tune λ in signed HMC-ECS, as explained in Section 5. All iterations used for tuning are included in the computational cost.

Figures 4 and 5 clearly show that the two HMC-ECS algorithms give better posterior approximations. The figures also show that SG-HMC with $L = 60$ leapfrog steps performs reasonably well in terms of accuracy and much better than SGLD, although some expectation estimates are biased. SG-HMC with $L = 7$ has substantially higher inefficiency factors and these figures show the degraded accuracy of the algorithm.

RCT	HMC-ECS$_{\text{S}}$	SG-HMC$_1$	SG-HMC$_2$	SGLD
min	1.2	6.8	48.6	53.9
median	1.6	9.5	100.2	230.1
max	2.6	682.3	246.7	2784.2

Table 3: Bankruptcy data. Relative computational time compared to perturbed HMC-ECS. For the two HMC-ECS algorithms, the cost is computed for the entire run including training and warmup period. The RCT with respect to stochastic gradient methods are based on post-burnin iteration only. HMC-ECS$_S$ is the signed HMC-ECS. SG-HMC$_1$ and SG-HMC$_2$ denote, respectively, the SG-HMC with $L = 60$ and $L = 7$ leapfrog steps.

	Perturbed HMC-ECS	Signed HMC-ECS
$\alpha_{\theta,\vec{p}}$	0.967	0.962
α_u	0.994	0.964
L	6	6
IF	2.202	2.31
ESS	912	871
$\widehat{\tau}$		1
$100(m/n)$	0.021	0.126

Table 4: Bankruptcy data. Summary of settings and efficiencies of HMC and HMC-ECS. The table shows the average acceptance probabilities (as a benchmark HMC has 0.966) in the post burn-in period for the two Gibbs steps, the number of steps L in the integrator used to obtain a predetermined trajectory length $\epsilon L = 1.2$, the average Ineffiency Factor (IF) (as a benchmark HMC has 2.195), the Effective Sample Size ESS = R/IF, the estimated probability of a positive likelihood estimator τ and the percentage of data used by each of the algorithms.

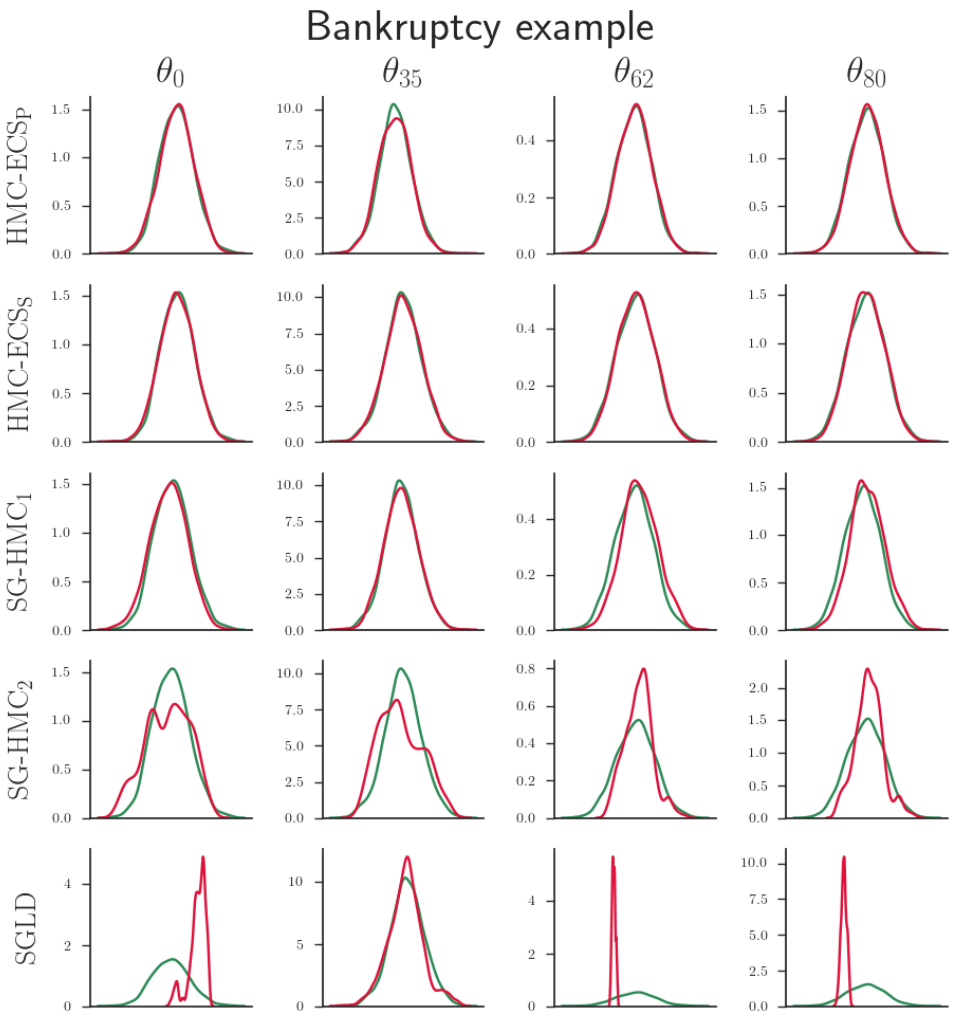

Figure 4: Bankruptcy data. The figure shows kernel density estimates of the posterior from the compared subsampling algorithms for four randomly selected parameters (green lines) and the corresponding posterior from HMC on the full data set (red lines). HMC-ECS$_P$ and HMC-ECS$_S$ denote, respectively, the perturbed and signed HMC-ECS$_S$. SG-HMC$_1$ and SG-HMC$_2$ denote, respectively, the SG-HMC with $L = 60$ and $L = 7$ leapfrog steps.

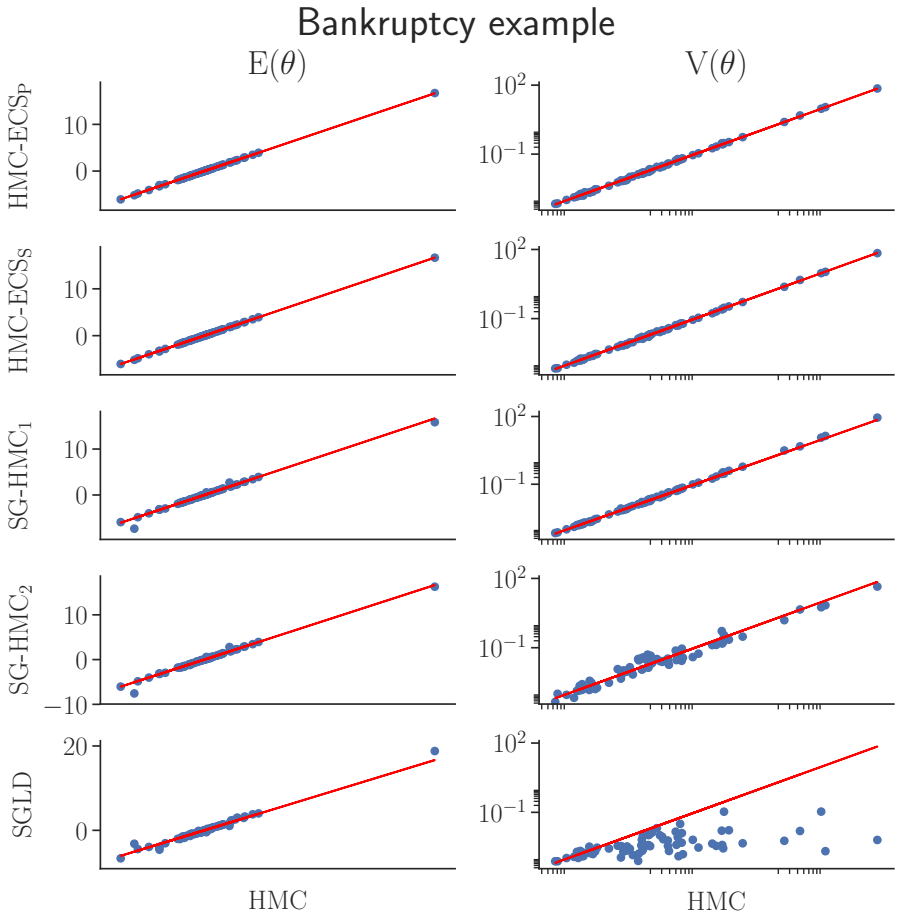

Figure 5: Bankruptcy data. The figure plots the estimated posterior mean and variance for all parameters from the subsampling algorithm against their true values obtained by HMC on the full data set. HMC-ECS$_{\text{P}}$ and HMC-ECS$_{\text{S}}$ denote, respectively, the perturbed and signed HMC-ECS$_{\text{S}}$. SG-HMC$_1$ and SG-HMC$_2$ denote, respectively, the SG-HMC with $L = 60$ and $L = 7$ leapfrog steps.

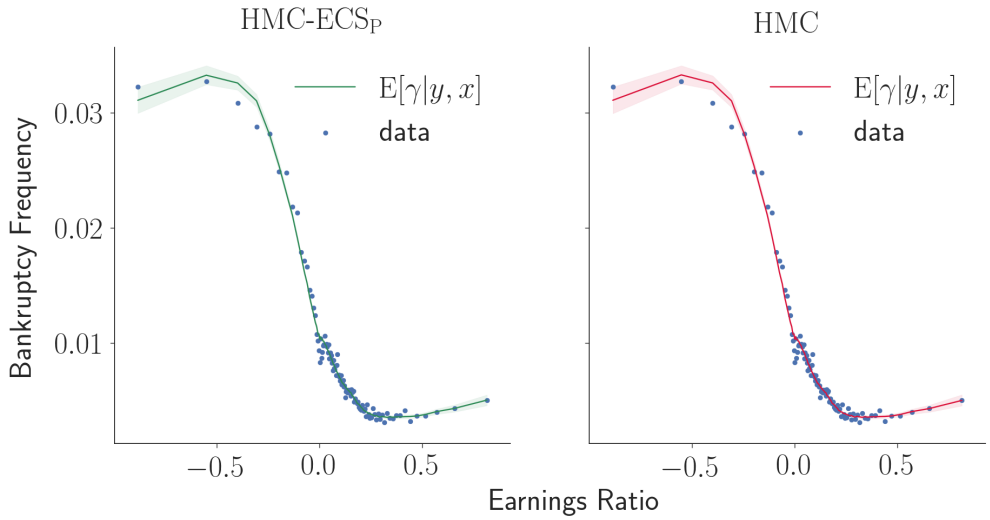

Figure 6: Realized and predicted bankruptcy probabilities as a function of the covariate *Earnings ratio* from the perturbed HMC-ECS (left panel) and HMC (right panel). Realized mean bankruptcy probabilities (blue dots) are computed by dividing the data into 100 equally sized groups based on the earnings ratio variable and estimating the bankruptcy probability by the fraction of bankrupt firms in each group. The solid line is the predictive mean and the shaded regions are point wise 90% equal tail posterior credible intervals.

Figure 6 shows the probability of bankruptcy for the fitted model and the empirical bankruptcy frequencies as a function of one of the covariates, Earnings ratio, with details in the caption. The posterior mean and posterior predictive intervals obtained by HMC and the perturbed HMC-ECS are indistinguishable.

Finally, we note that implementing HMC using the full data in this example is, for the perturbed and exact approach respectively, 478.7 and 311.5 times more expensive in terms of posterior density and gradient evaluations.

6.7. Scalability of HMC-ECS

Beskos et al. (2013) show that, in an optimally tuned HMC algorithm, the step size ϵ needs to be scaled as $O(d^{-1/4})$ to keep the acceptance probability constant as the dimension d increases. This is more favorable than the rate $O(d^{-1/3})$ of Langevin Monte Carlo (Roberts and Rosenthal, 1998). We have argued that since our algorithm is performing a HMC step using a Hamiltonian based on a subset of the data it should scale with dimension similarly to HMC.

We set out to test this hypothesis empirically as follows. First, we consider a sequence of d, obtained as $d = 2^h$, $h = 1, \ldots, 8$ and obtain eight simulated data sets with $n = 10,000$ each. For each d, we run the dual averaging algorithm as described in Section 6.3 to find the optimal $\epsilon$and check if $\epsilon = O(d^{-1/4})$ is reasonable. In agreement with Beskos

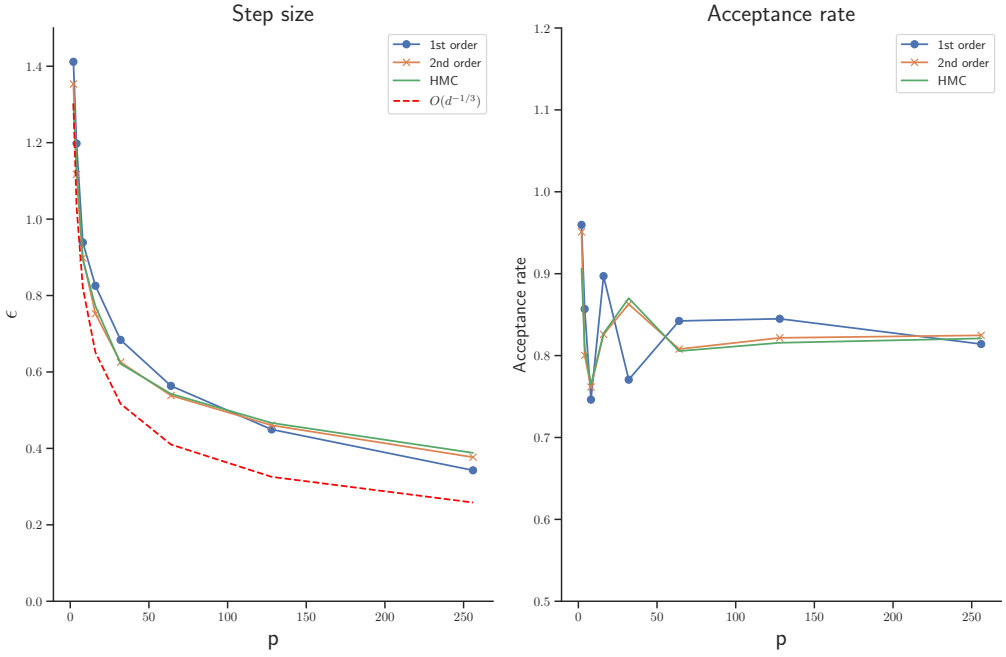

Figure 7: Empirical illustration of the scaling of HMC-ECS, see Section 6.7 for the experimental settings. The left figure shows the optimal step size as a function of the dimension. HMC ($O(d^{-1/4})$) and Langevin ($O(d^{-1/3})$) are plotted for reference. The right figure shows the acceptance rate targeted for optimality. The IF of our methods are close to 1, regardless of if the 1st or 2nd control variate is used. The variance of the log-likelihood estimator for the second order control variate is nearly zero (quadratic target) and for the first order control variate is kept around 1 (Pitt et al., 2012) by selecting m appropriately.

et al. (2013) we set M optimally from the curvature of the conditional target posterior. This is easily achieved by considering a Gaussian regression model where we set the prior $p_\Theta(\theta) = \mathcal{N}(0, 5^2 I_d)$, such that the optimal M for the conditional target is, assuming the bias-correction term to be negligible,

$$M = \frac{n}{m} \sum_{i=1}^{m} \mathbf{X}_{u_i}^\top \mathbf{X}_{u_i} + \frac{1}{5^2},$$

where $\mathbf{X}_{u_i}$ denotes the u_ith row of the design matrix. We note that, since the Gaussian model is quadratic in its log-density, the second order control variate will yield a perfect fit, i.e. the variance of $\widehat{l}_m(\theta)$ is zero. Thus, we also experiment with a first order control variate. We scale m to maintain the variance around 1 (Pitt et al., 2012). Figure 7 shows the results and we deduce that the algorithm does indeed maintain scalability.

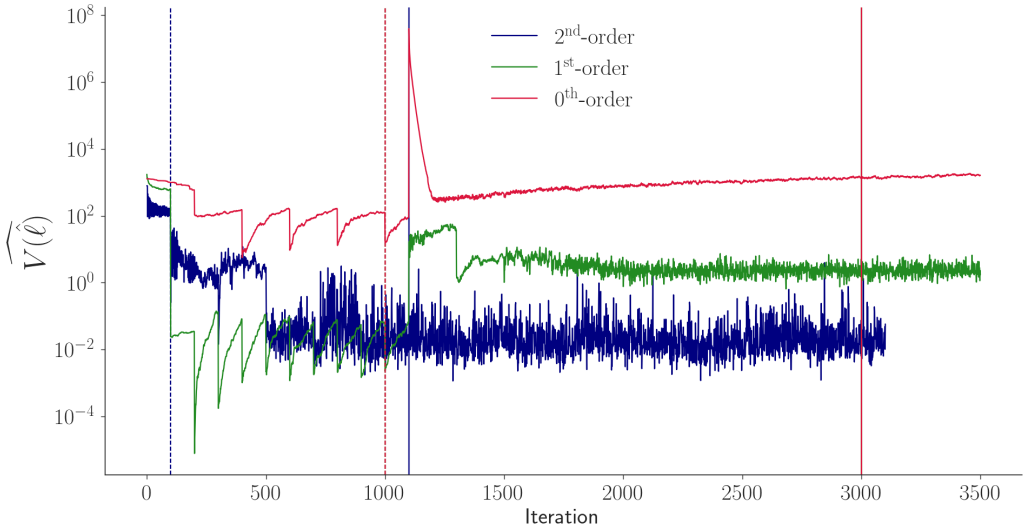

Figure 8: Variance of $\widehat{l}_m$. The figure shows the estimated variance as a function of the iterations for three different orders in the Taylor expansion for the control variates. The dashed vertical lines correspond to the end of the training period used the solid vertical line corresponds to the end of the burn-in period.

6.8. Limitations of subsampling HMC

Variance reduction by control variates is crucial in any subsampling MCMC algorithm. This subsection explores the role of control variates by successively degrading the quality of the control variates by lowering the order of their Taylor approximation. Figure 8 shows the estimated variance of $\widehat{l}_m(\theta)$ as a function of the iterates when the control variates are based on a Taylor series expansions of different orders. We note that the algorithm survives a substantial variance during the training iterations when applying the first order control variates, and this variance eventually settles down once a sensible $\theta^\star$ is found. Figure 9 shows that the expectation and variance estimates are very accurate for the first order control variates. Figures 8 and 9 also shows that control variates of zero order are too crude for HMC-ECS in this example. While it seems that the competing methods are more robust to the quality of the control variates, we again stress that all competitors are placed in the unrealistically favorable scenario of having M equal to the inverse Hessian evaluated at the posterior mean.

The control variates (Bardenet et al., 2017) used in this paper are quadratic in θ centered around $\theta^\star$. This means that they are expected to work well for any model which has a log-density that is reasonably quadratic in θ in a neighborhood of $\theta^\star$. Examples are Poisson regression (Quiroz et al., 2018c) and Student-t regression (Quiroz et al., 2019). While the locally approximately quadratic feature is found in many models, there are clearly models in which this local approximation can be poor, for example deep neural nets. To find efficient control variates, which are also computationally feasible, in a class of complex models remains an open challenging problem. We stress however that this is a problem for

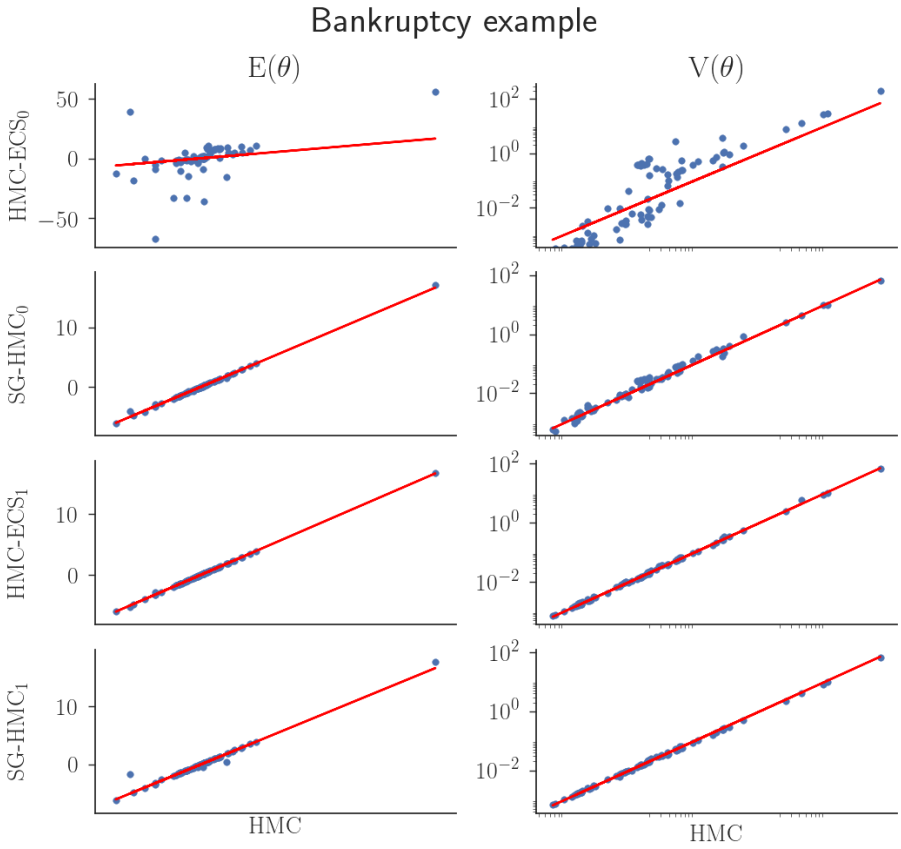

Figure 9: Results for the zeroth and first order control variate. The figure shows the estimates of posterior expectations and posterior variances using the zeroth order control variate (upper panel) and first order control variate (lower panel). All comparisons are versus HMC which represents the ground truth.

all existing subsampling MCMC approaches, and any progress on improved control variates can be straightforwardly incorporated into HMC-ECS.

7. Conclusions and Future Research

We propose a method to speed up Bayesian inference while maintaining high sampling efficiency in moderately high-dimensional parameter spaces by combining data subsampling and Hamiltonian Monte Carlo such that the energy is conserved. We show how to implement the method using two estimators of the likelihood. The first implementation, which we refer to as perturbed HMC-ECS, produces iterates from a perturbed density that will get arbitrarily close to the true density, as measured by the total variation metric, at the rate $O(n^{-1}m^{-2})$. The second implementation, which we refer to as signed HMC-ECS, gives iterates which are then used in an importance sampling estimator to obtain a simulation consistent estimator of the expectation of any posterior functional.

We apply the methods to simulate from the posterior distribution in two data sets, with $d = 29$ and $d = 81$ dimensions, respectively. Our two HMC-ECS algorithms perform highly accurate inference, comparable to HMC without subsampling, but are computationally much faster. This is a major step forward since Bardenet et al. (2017) and Quiroz et al. (2019) demonstrate that most subsampling approaches cannot even beat standard MH without subsampling on toy examples with $d = 2$ and highly redundant data. We also show that HMC-ECS is very competitive against SGLD and SG-HMC, both in terms of sampling efficiency and accuracy.

Control variates to reduce the variance of subsampling estimators are well known to be crucial for any subsampling MCMC algorithm. We use very efficient control variates based on a second order Taylor expansion in our applications, but explore the effects of less accurate control variates. We find that HMC-ECS still performs well with a cruder first order approximation, but that a Taylor approximation of order zero is too crude and gives a too large variance for HMC-ECS.

Similarly to HMC, HMC-ECS is difficult to tune. Self-tuning algorithms such as the no-U-Turn sampler (Hoffman and Gelman, 2014) have been proposed for HMC and it would be interesting to see if our ideas can be applied there. It would also be interesting to consider Riemann Manifold HMC (Girolami and Calderhead, 2011), which has been demonstrated to be very effective when a high-dimensional posterior exhibits strong correlations. Scaling up such an algorithm opens up the possibility of simulating the posterior density of highly complex models with huge data sets. Finally, until recently, one of the limitations of HMC was its inability to cope with discrete parameters. Nishimura et al. (2017) overcomes this limitation and extending HMC-ECS in this direction would be an interesting undertaking.

Acknowledgments

We thank the action editor and two reviewers for constructive comments that improved both the content and presentation of the paper.

Matias Quiroz and Robert Kohn were partially supported by Australian Research Council Center of Excellence grant CE140100049. Mattias Villani was partially supported by Swedish Foundation for Strategic Research (Smart Systems: RIT 15-0097).

References

Christophe Andrieu and Gareth O Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725, 2009.

Jack Baker, Paul Fearnhead, Emily B Fox, and Christopher Nemeth. Control variates for stochastic gradient MCMC. *arXiv preprint arXiv:1706.05439*, 2017.

Pierre Baldi, Peter Sadowski, and Daniel Whiteson. Searching for exotic particles in high-energy physics with deep learning. *Nature Communications*, 5(4308), 2014.

Rémi Bardenet, Arnaud Doucet, and Chris Holmes. Towards scaling up Markov chain Monte Carlo: an adaptive subsampling approach. In *Proceedings of The 31st International Conference on Machine Learning*, pages 405–413, 2014.

Rémi Bardenet, Arnaud Doucet, and Chris Holmes. On Markov chain Monte Carlo methods for tall data. *Journal of Machine Learning Research*, 18(47):1–43, 2017.

José M Bernardo and Adrian FM Smith. *Bayesian Theory*. John Wiley & Sons, 2001.

Alexandros Beskos, Natesh Pillai, Gareth Roberts, Jesus-Maria Sanz-Serna, and Andrew Stuart. Optimal tuning of the hybrid Monte Carlo algorithm. *Bernoulli*, 19(5A):1501–1534, 2013.

Michael Betancourt. The fundamental incompatibility of scalable Hamiltonian Monte Carlo and naive data subsampling. In *International Conference on Machine Learning*, pages 533–540, 2015.

Michael Betancourt. A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*, 2017.

Joris Bierkens, Paul Fearnhead, and Gareth Roberts. The zig-zag process and super-efficient sampling for Bayesian analysis of big data. *The Annals of Statistics*, 47(3):1288–1320, 2019.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

Nicolas Brosse, Alain Durmus, and Eric Moulines. The promises and pitfalls of stochastic gradient Langevin dynamics. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 8277–8287. Curran Associates, Inc., 2018.

DM Ceperley and M Dewing. The penalty method for random walks with uncertain energies. *The Journal of Chemical Physics*, 110(20):9812–9820, 1999.

Niladri Chatterji, Nicolas Flammarion, Yian Ma, Peter Bartlett, and Michael Jordan. On the theory of variance reduction for stochastic gradient Monte Carlo. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 764–773. PMLR, 2018.

Tianqi Chen, Emily Fox, and Carlos Guestrin. Stochastic gradient Hamiltonian Monte Carlo. In *International Conference on Machine Learning*, pages 1683–1691, 2014.

George Deligiannidis, Arnaud Doucet, and Michael K Pitt. The correlated pseudomarginal method. *Journal of the Royal Statistical Society. Series B (Methodological)*, 80(5):839–870, 2018.

Arnaud Doucet, MK Pitt, George Deligiannidis, and Robert Kohn. Efficient implementation of Markov chain Monte Carlo when using an unbiased likelihood estimator. *Biometrika*, 102(2):295–313, 2015.

Simon Duane, Anthony D Kennedy, Brian J Pendleton, and Duncan Roweth. Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222, 1987.

Kumar Avinava Dubey, Sashank J Reddi, Sinead A Williamson, Barnabas Poczos, Alexander J Smola, and Eric P Xing. Variance reduction in stochastic gradient Langevin dynamics. In *Advances in Neural Information Processing Systems*, pages 1154–1162, 2016.

Paolo Giordani, Tor Jacobson, Erik von Schedvin, and Mattias Villani. Taking the twists into account: Predicting firm bankruptcy risk with splines of financial ratios. *Journal of Financial and Quantitative Analysis*, 49(4):1071–1099, 2014.

Mark Girolami and Ben Calderhead. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(2):123–214, 2011.

David Gunawan, Robert Kohn, Matias Quiroz, Khue-Dung Dang, and Minh-Ngoc Tran. Subsampling sequential Monte Carlo for static Bayesian models. *arXiv preprint arXiv:1805.03317*, 2018.

W Keith Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

Matthew D Hoffman and Andrew Gelman. The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(1): 1593–1623, 2014.

Pierre E Jacob and Alexandre H Thiery. On nonnegative unbiased estimators. *The Annals of Statistics*, 43(2):769–784, 2015.

Anoop Korattikara, Yutian Chen, and Max Welling. Austerity in MCMC land: Cutting the Metropolis-Hastings budget. In *Proceedings of the 31st International Conference on Machine Learning*, pages 181–189, 2014.

Fredrik Lindsten and Arnaud Doucet. Pseudo-marginal Hamiltonian Monte Carlo. *arXiv preprint arXiv:1607.02516*, 2016.

Anne-Marie Lyne, Mark Girolami, Yves Atchade, Heiko Strathmann, and Daniel Simpson. On Russian roulette estimates for Bayesian inference with doubly-intractable likelihoods. *Statistical Science*, 30(4):443–467, 2015.

Yi-An Ma, Tianqi Chen, and Emily Fox. A complete recipe for stochastic gradient MCMC. In *Advances in Neural Information Processing Systems*, pages 2917–2925, 2015.

Dougal Maclaurin and Ryan P Adams. Firefly Monte Carlo: Exact MCMC with subsets of data. In *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence (UAI 2014)*., 2014.

Florian Maire, Nial Friel, and Pierre Alquier. Informed sub-sampling MCMC: approximate Bayesian inference for large datasets. *Statistics and Computing*, To appear, 2018.

Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

Stanislav Minsker, Sanvesh Srivastava, Lizhen Lin, and David Dunson. Scalable and robust Bayesian inference via the median posterior. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1656–1664, 2014.

Radford M. Neal. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, 2(11):113–162, 2011.

Willie Neiswanger, Chong Wang, and Eric Xing. Asymptotically exact, embarrassingly parallel MCMC. *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence (UAI 2014)*, pages 623–632, 2014.

Christopher Nemeth and Chris Sherlock. Merging MCMC subposteriors through Gaussian-process approximations. *Bayesian Analysis*, 13(2):507–530, 2018.

Christopher Nemeth, Chris Sherlock, and Paul Fearnhead. Particle Metropolis-adjusted Langevin algorithms. *Biometrika*, 103(3):701–717, 2016.

Geoff K Nicholls, Colin Fox, and Alexis Muir Watt. Coupled MCMC with a randomized acceptance probability. *arXiv preprint arXiv:1205.6857*, 2012.

Akihiko Nishimura, David Dunson, and Jianfeng Lu. Discontinuous Hamiltonian Monte Carlo for sampling discrete parameters. *arXiv preprint arXiv:1705.08510*, 2017.

Michael K Pitt, Ralph dos Santos Silva, Paolo Giordani, and Robert Kohn. On some properties of Markov chain Monte Carlo simulation methods based on the particle filter. *Journal of Econometrics*, 171(2):134–151, 2012.

Martyn Plummer, Nicky Best, Kate Cowles, and Karen Vines. CODA: convergence diagnosis and output analysis for MCMC. *R News*, 6(1):7–11, 2006.

Matias Quiroz, Minh-Ngoc Tran, Mattias Villani, and Robert Kohn. Speeding up MCMC by delayed acceptance and data subsampling. *Journal of Computational and Graphical Statistics*, 27(1):12–22, 2018a.

Matias Quiroz, Minh-Ngoc Tran, Mattias Villani, Robert Kohn, and Khue-Dung Dang. The block-Poisson estimator for optimally tuned exact subsampling MCMC. *arXiv preprint arXiv:1603.08232v5*, 2018b.

Matias Quiroz, Mattias Villani, Robert Kohn, Minh-Ngoc Tran, and Khue-Dung Dang. Subsampling MCMC - A introduction for the survey statistician. *Sankhya A*, 80(1): 33–69, 2018c.

Matias Quiroz, Robert Kohn, Mattias Villani, and Minh-Ngoc Tran. Speeding up MCMC by efficient data subsampling. *Journal of the American Statistical Association*, 114(526): 831–843, 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, 1998.

Gareth O Roberts and Jeffrey S Rosenthal. General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1:20–71, 2004.

Gareth O Roberts and Osnat Stramer. Langevin diffusions and Metropolis-Hastings algorithms. *Methodology and Computing in Applied Probability*, 4(4):337–357, 2002.

Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 71(2):319–392, 2009.

Sebastian M Schmon, George Deligiannidis, Arnaud Doucet, and Michael K Pitt. Large sample asymptotics of the pseudo-marginal method. *arXiv preprint arXiv:1806.10060*, 2018.

Steven L Scott, Alexander W Blocker, Fernando V Bonassi, H Chipman, E George, and R McCulloch. Bayes and big data: the consensus Monte Carlo algorithm. In *EFaBBayes 250" conference*, volume 16, 2013.

Xiaocheng Shang, Zhanxing Zhu, Benedict Leimkuhler, and Amos J Storkey. Covariance-controlled adaptive Langevin thermostat for large-scale Bayesian sampling. In *Advances in Neural Information Processing Systems*, pages 37–45, 2015.

Chris Sherlock, Alexandre H Thiery, Gareth O Roberts, and Jeffrey S Rosenthal. On the efficiency of pseudo-marginal random walk Metropolis algorithms. *The Annals of Statistics*, 43(1):238–275, 2015.

Yee Whye Teh, Alexandre H Thiery, and Sebastian J Vollmer. Consistency and fluctuations for stochastic gradient Langevin dynamics. *Journal of Machine Learning Research*, 17: 1–33, 2016.

M-N Tran, R Kohn, M Quiroz, and M Villani. The block pseudo-marginal sampler. *arXiv preprint arXiv:1603.02485v5*, 2017.

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.

Xiangyu Wang and David B Dunson. Parallel MCMC via Weierstrass sampler. *arXiv preprint arXiv:1312.4605v2*, 2014.

Max Welling and Yee W Teh. Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th International Conference on Machine Learning*, pages 681–688, 2011.