Divide-and-conquer Metropolis-Hastings Samplers with Matched Samples

Hang Qian*

Abstract. Divide-and-conquer methods for scalable Bayesian inference divide the massive data into subsets, sample from the subset posterior distributions, and then combine the results. We develop an asymptotically exact recombination method by matched samples. Subset posterior densities calculated by the Metropolis-Hastings samplers are recycled for evaluating the importance weight to reduce the computational burden. Our computationally efficient aggregation algorithm features a collection of consistent estimators of expectations with respect to the full posterior distribution. Weight degeneracy of the importance sampling is resolved by the matched-sample resample-move method, which handles heterogeneous and non-overlapping subposteriors. Numeric examples and real-world mortgage data applications demonstrate excellent performance of the novel approach.

Keywords: Bayes, Big data, Markov chain Monte Carlo, Parallel computing.

1 Introduction

Posterior simulation by the Metropolis-Hastings (MH) sampler (Metropolis et al., 1953; Hastings, 1970) is computationally intensive or infeasible if the number of observations is large. It is tempting to split the massive data into subsets, run the MH sampler on each subset separately, and then merge the results. The open question for this approach is how to combine the Markov chain Monte Carlo (MCMC) samples for approximating the full posterior, as Bardenet et al. (2017) comment that 'divide-and-conquer approaches have yet to solve the recombination problem, i.e. how to obtain a meaningful distribution in a stable manner from the output of individual chains on a growing number of smaller datasets.'

The divide-and-conquer methods proposed in the literature vary in the assumptions on the subset posterior distributions (abbreviated as subposteriors). Scott et al. (2016) recombine subposterior samples in a way that would be exact if the subposteriors had been Gaussian. The attraction is a closed-form formula for merging subposteriors at the lowest computing cost. Neiswanger et al. (2014) propose a non-parametric merging algorithm, in which subposteriors are approximated by the Gaussian kernel density estimator, so that the merged posterior follows a Gaussian mixture distribution. Wang et al. (2015) assume that each subposterior is approximated by a K-block histogram. Wang and Dunson (2013) approximate the subposterior densities via the Weierstrass transformation. It is also possible to replicate subset data multiple times, which raises the likelihood in every subset to a power. The boosted subposteriors are combined through their geometric center (Minsker et al., 2017; Srivastava et al., 2018). Compared

*55 Centre Street, Natick, MA 01760, USA.

to previous approaches that solely rely on subposterior MCMC samples, the insight of Nemeth and Sherlock (2018) is to utilize the subposterior densities calculated in the MH sampler. They assume that the log subposterior density follows a Gaussian process, which converts the limited information on a finite set of points to information on the whole support of the subposterior density.

We follow the idea of Nemeth and Sherlock (2018) and recycle the subposterior densities for estimating the full posterior expectations. The main difference is that we avoid distributional assumptions on the subposteriors. The key contribution of our paper is an asymptotically exact divide-and-conquer MCMC method that combines subposterior samples using importance weighting. The weight evaluations, however, involve recycled subposterior densities that save the computational cost.

Our idea is to share global proposals between subposterior chains. Local proposals for subposterior MCMC samplers are obtained by rejection sampling from global proposals. Each subposterior sample has an exact match in global proposals, hence the name *matched samples*. When a subposterior density is queried by the importance sampler, it is available as a by-product of the MH sampler. That is, subposterior densities are recycled for evaluating the importance weight to reduce the computational burden.

Weight degeneracy undermines the importance sampling. Gilks and Berzuini (2001) restore particle diversity by the resample-move method using a Markov transition kernel. We adapt the resample-move method to the divide-and-conquer algorithm that accommodates heterogeneous and non-overlapping subposteriors. Our method addresses weight degeneracy at a low computing cost.

Another stream of research on the big data Bayesian inference investigates subsampling by randomly selecting a subset data at each iteration of the MCMC sampler. Various subsampling schemes have been proposed in the literature: firefly Monte Carlo by Maclaurin and Adams (2014), adaptive subsampling with MH tests by Korattikara et al. (2014), stochastic gradient Langevin dynamics by Teh et al. (2016), bias-corrected subsampling with control variates by Quiroz et al. (2019), zig-zag process with subsampling by Bierkens et al. (2019), MH sampling with delayed acceptance by Banterle et al. (2019), among others. Bardenet et al. (2017) provide an excellent review. Although subsampling reduces data points for evaluating the likelihood function, it requires the whole data loaded in memory at all times. In contrast, it is possible to apply our method when data are saved across multiple machines for parallel processing.

The remainder of the paper is structured as follows. Section 2 integrates MCMC, importance and rejection sampling to estimate posterior expectations by matched samples. Section 3 addresses weight degeneracy by the resample-move strategy. Section 4 evaluates the performance of our algorithm by numeric examples, including non-overlapping subposteriors, heterogeneity induced by unidentified parameters in some subset data and multimodal posterior distributions. Section 5 applies our method to the real-world mortgage data. Instead of random partition, each subset consists of the loan data from a specific lender, and each subposterior distribution has its own economic interpretation. Section 6 concludes the paper and discusses directions of future research.

2 Divide and Conquer by Matched Samples

Let Y be the data and $p(Y|\theta)$ be the likelihood, where θ is the model parameters with the prior $p(\theta)$. If the number of observations is moderate, we may run the MH sampler targeting the full posterior

$$p(\theta|Y) \propto p(\theta)\, p(Y|\theta).$$

If the data are too large to be stored or processed by a single machine, a natural solution is to divide the data into subsets $Y_1, \ldots, Y_m$ such that

$$p(Y|\theta) = \prod_{j=1}^{m} p(Y_j|\theta),$$

and run the MH sampler targeting the subposterior

$$p(\theta|Y_j) \propto p(\theta)^{1/m}\, p(Y_j|\theta),$$

for $j = 1, \ldots, m$ separately, possibly with distributed and parallel computing.

To merge the subposteriors, we resort to Equation (1), in which the full posterior is proportional to the product of subposteriors:

$$p(\theta|Y) \propto \prod_{j=1}^{m} p(\theta|Y_j). \tag{1}$$

An importance sampling method decomposes Equation (1) as

$$p(\theta|Y) \propto p(\theta|Y_j)\, \tilde{w}_j(\theta), \tag{2}$$

where $\tilde{w}_j(\theta) = \prod_{i \neq j} p(\theta)^{1/m}\, p(Y_i|\theta)$ is the unnormalized importance weight. We have MCMC samples (i.e., particles) from $p(\theta|Y_j)$, and the weighted samples can approximate the full posterior expectations. The importance sampling is applicable to each subset $j = 1, \ldots, m$ separately, which provides m alternative estimators and they can validate each other.

The method has pros and cons. Scott (2017) rates it as "the most theoretically pure of the methods we consider", as it makes no assumptions on the shape of the subposterior distributions being combined, and the self-normalized importance sampling produces consistent estimates of expectations with respect to the full posterior. However, the method has two shortcomings. First, it can be computationally intensive to evaluate the importance weight. Scott (2017) suggests that each subposterior sampler broadcasts its output particles to other samplers for evaluating the likelihood function. If each sampler generates T draws, it amounts to $(m-1)\,mT$ likelihood evaluations that require $(m-1)\,T$ passes of the massive data, which can be a computational burden. Second, the importance sampling is plagued by weight degeneracy if $p(\theta|Y_j)$ is substantially different from $p(\theta|Y)$. A small number of particles carry most of the weight, while most particles do not significantly contribute to the full posterior approximation.

We develop a novel approach that inherits merits of the importance sampling, and alleviates the computational burden and weight degeneracy. Our idea is to use the matched samples in the subposterior simulation, and recycle the subposterior densities for evaluating the importance weight. To support matched samples, we redesign the workflow of the standard MH sampler (Algorithm 1).

Algorithm 1 (Standard MH sampler)**.**

Input*: subset data* Y_j *and proposal transition kernel* $q_j(\theta^*|\theta)$*.*

Output*: MCMC samples* $\{\theta_{jt}\}$ *from the subposterior* $p(\theta|Y_j)$*.*

1 *Specify an initial value* θ_{j0} *for the subset Markov chain.*

2 *Iterate for* $t = 1, 2, \ldots, T$*:*

2.1 *Generate* θ^* *by* $q_j(\theta^*|\theta)$*, where* $\theta = \theta_{j,t-1}$*.*

2.2 *Compute* $\alpha = \min\left[1, \frac{p(\theta^*)^{1/m} p(Y_j|\theta^*) q_j(\theta|\theta^*)}{p(\theta)^{1/m} p(Y_j|\theta) q_j(\theta^*|\theta)}\right]$*.*

2.3 *Set* $\theta_{jt} = \begin{cases} \theta^* & \text{with probability } \alpha \\ \theta & \text{with probability } 1-\alpha \end{cases}$*.*

We refer to $q_j(\theta^*|\theta)$ as the local proposal density for the j^{th} subposterior MCMC. Instead of direct sampling from the local proposal density, we specify a global proposal density $q(\theta^*)$ for generating local proposal draws by rejection sampling, so that each local proposal draw has an exact match in global proposals. For example, $q(\theta^*)$ and $q_j(\theta^*|\theta)$ can be Gaussian densities $\phi(\theta^*; \mu, \Sigma)$ and $\phi(\theta^*; \mu_j(\theta), \Sigma_j)$ respectively. A global proposal from the former is accepted by the probability

$$\frac{\phi(\theta^*; \mu_j, \Sigma_j)}{\phi(\theta^*; \mu, \Sigma) B_j},$$

where $\mu_j(\theta)$ is abbreviated as μ_j, and the upper bound of the density ratio is given by

$$B_j = \frac{|\Sigma|^{1/2}}{|\Sigma_j|^{1/2}} e^{-\frac{1}{2}\left[\mu_j' \Sigma_j^{-1} \mu_j - \mu' \Sigma^{-1} \mu - \left(\Sigma_j^{-1} \mu_j - \Sigma^{-1} \mu\right)' \left(\Sigma_j^{-1} - \Sigma^{-1}\right)^{-1} \left(\Sigma_j^{-1} \mu_j - \Sigma^{-1} \mu\right)\right]}. \quad (3)$$

A special case of the global-local proposal specification is $q_j(\theta^*|\theta) = q(\theta^*)$, which leads to an independence MH sampler that the transition kernel does not depend on the current state of the Markov chain. In that case, all global proposals are accepted as local proposals. All proposals for the MH sampler are generated independently before the first iteration of the MCMC sampling, instead of making one proposal at each iteration.

Regarding efficient design of global and local proposal distributions, we often face a tradeoff between the efficiency of the rejection sampling and the MH sampling acceptance rates, especially when the subposterior distributions are centered far away from each other. Sampler tuning is problem specific. In the numeric examples in Section 4,

we show three design patterns: 1) $\Sigma > \Sigma_j$ and μ_j is close to the subposterior mean, 2) $\Sigma = \Sigma_j$, which is obtained from the inverse Hessian by numerical optimization, and 3) $\Sigma > \Sigma_j$ and $\mu_j(\theta) = \theta$, as in the random-walk MH sampler.

In Algorithm 2, we denote global proposals by $\theta^*_{(n)}$, $n = 1, \ldots, N$, local proposals by θ^*_{jt} and MCMC samples by θ_{jt}, $t = 1, \ldots, T$. For example, if $\theta^*_{(1)}, \theta^*_{(3)}, \theta^*_{(5)}, \ldots$ are accepted by rejection sampling for the j^{th} subposterior simulation, we denote local proposals as $\theta^*_{j1} = \theta^*_{(1)}$, $\theta^*_{j2} = \theta^*_{(3)}$, $\theta^*_{j3} = \theta^*_{(5)}$, and so on. Because all MCMC samples are members of local proposals, and all local proposals are members of global proposals, we refer to them as matched samples.

Algorithm 2 (MH sampler with matched samples)**.**

Input*: subset data Y_j, proposal densities $q(\cdot)$ and $q_j(\cdot)$.*

Output*: MCMC samples $\{\theta_{jt}\}$ from the subposterior $p(\theta|Y_j)$.*

1 *Generate global proposals $\theta^*_{(n)}$, $n = 1, \ldots, N$ by $q(\cdot)$.*

2 *Specify an initial value θ_{j0} for the subset Markov chain.*

3 *Iterate for $t = 1, \ldots, T$:*

3.1 *Select θ^*_{jt} from $\theta^*_{(n)}$ by rejection sampling with the target $q_j(\cdot)$.*

3.2 *Compute* $\alpha = \min\left[1, \frac{p(\theta^*_{jt})^{1/m} p(Y_j|\theta^*_{jt}) q_j(\theta|\theta^*_{jt})}{p(\theta)^{1/m} p(Y_j|\theta) q_j(\theta^*_{jt}|\theta)}\right]$*, where* $\theta = \theta_{j,t-1}$.

3.3 *Set* $\theta_{jt} = \begin{cases} \theta^*_{jt} & \text{with probability } \alpha \\ \theta & \text{with probability } 1-\alpha \end{cases}$.

Algorithm 2 is applied to each subposterior sampler separately without communication. The matched samples can be created by fixing the random number seed for generating the global proposals $\theta^*_{(n)}$, $n = 1, \ldots, N$, This is equivalent to broadcasting the global proposals to all samplers before the first iteration of the MH sampling. The MH samplers evaluate the likelihood function $p\left(Y_j \middle| \theta^*_{jt}\right)$, $j = 1, \ldots, m$ and $t = 1, \ldots, T$. This is the output that we want to recycle.

Regarding the choice of N and T, the upper bound of the global-local density ratio B_j in Equation (3) determines the ratio of N and T. If the local proposal $q_j(\theta^*|\theta)$ does not depend on θ, approximately $N = \max_j B_j T$ is needed. If the local proposal depends on θ, the upper bound B_j varies at each iteration of MCMC sampling. In practice, global proposals are generated under a common random number seed for all subposterior samplers, so N can be dynamically determined during subposterior sampling, given a user-specified T. Alternatively, it is possible to pre-specify a hyperparameter N and resample from $\left\{\theta^*_{(1)}, \ldots, \theta^*_{(N)}\right\}$ as the global proposals. Asymptotically, a resampled global proposal still follows $q(\cdot)$.

Algorithm 3 summarizes the divide-and-conquer MCMC and importance sampling with the matched samples. The novelty is recycled densities in the importance weight

at step 4. Because θ_{jt} is a member of the global proposals, it is likely that the likelihood function $p(Y_i|\theta_{jt})$ has been evaluated by the i^{th} subposterior sampler, and can be recycled for the importance weight evaluation.

Algorithm 3 (Divide and conquer by matched samples)**.**
Input*: data Y, proposal densities $q(\cdot)$ and $q_j(\cdot)$, and a function of interest $h(\theta)$.*
Output*: a collection of estimators of $\int h(\theta)\, p(\theta|Y)\, d\theta$.*

1 *Divide data Y into subsets $Y_1, \ldots, Y_m$.*

2 *Generate global proposals $\theta^*_{(n)}$, $n = 1, \ldots, N$ by $q(\cdot)$.*

3 *Share global proposals and apply Algorithm 2 to each subposterior sampler.*

4 *Compute unnormalized and normalized importance weight*

$$\tilde{w}_j(\theta_{jt}) = p(\theta_{jt})^{\frac{m-1}{m}} \prod_{i \neq j} p(Y_i|\theta_{jt}),$$

$$w_j(\theta_{jt}) = \frac{\tilde{w}_j(\theta_{jt})}{\sum_{\tau=1}^{T} \tilde{w}_j(\theta_{j\tau})}.$$

5 *Approximate $\int h(\theta)\, p(\theta|Y)\, d\theta$ by $\sum_{t=1}^{T} h(\theta_{jt})\, w_j(\theta_{jt})$, $j = 1, \ldots, m$.*

The recycle rate is maximized if we specify $q_j(\theta^*|\theta) = q(\theta^*)$, $\forall j$, so that the importance weight is fully determined by the recycled densities. Data management is convenient in that case. Step 1 of Algorithm 3 does not require random or balanced partition, and the subset data can be stored on different machines. Data are accessed at step 3, which can be performed in parallel. When the same set of global and local proposals are applied to each subposterior chain, calculating the importance weight in step 4 does not require scanning the original data again, and therefore step 4 takes negligible computing time relative to subposterior sampling.

The feature of sample matching does not add complexity to the asymptotic analysis, if we separately examine each subposterior Markov chain underlying the MH sampler. For an ergodic Markov chain, the sample average converges to the expected value. Weighted particles can consistently estimate expectations with respect to the full posterior. Proposition 1 shows that Algorithm 3 provides m consistent estimators of $\int h(\theta)\, p(\theta|Y)\, d\theta$ under mild regularity conditions.

Proposition 1. *Consider the j^{th} MH sampler targeting the subposterior $p(\theta|Y_j)$ in Algorithm 3. Suppose that the MCMC samples θ_{jt}, $t = 1, \ldots, T$ are generated by an irreducible, aperiodic and Harris recurrent Markov chain, $\int |h(\theta)\, \tilde{w}_j(\theta)|\, p(\theta|Y)\, d\theta < \infty$ and $\int |\tilde{w}_j(\theta)|\, p(\theta|Y)\, d\theta < \infty$. As $T \to \infty$, we have*

$$\sum_{t=1}^{T} h(\theta_{jt})\, w_j(\theta_{jt}) \xrightarrow{p} \int h(\theta)\, p(\theta|Y)\, d\theta.$$

Proof. We write Equation (2) in a verbose form:

$$p\left(\theta\left|Y\right.\right)=p\left(\theta\left|Y_{j}\right.\right)\tilde{w}_{j}\left(\theta\right)\frac{p\left(Y_{j}\right)k}{p\left(Y\right)},$$

where $k=\int p\left(\theta\right)^{1/m}d\theta$, $p\left(Y_{j}\right)=\frac{1}{k}\int p\left(\theta\right)^{1/m}p\left(Y_{j}\left|\theta\right.\right)d\theta$, and $p\left(Y\right)=\int p\left(\theta\right)p\left(Y\left|\theta\right.\right)d\theta$. To clarify our notation, $p\left(\theta\left|Y_{j}\right.\right)$ and $p\left(Y_{j}\right)$ refer to the posterior and marginal likelihood under the prior $\frac{1}{k}p\left(\theta\right)^{1/m}$, rather than $p\left(\theta\right)$, used in the j^{th} subposterior MCMC.

The weighted sample average is

$$\sum_{t=1}^{T}h\left(\theta_{jt}\right)w_{j}\left(\theta_{jt}\right)=\frac{T^{-1}\sum_{t=1}^{T}h\left(\theta_{jt}\right)\tilde{w}_{j}\left(\theta_{jt}\right)}{T^{-1}\sum_{t=1}^{T}\tilde{w}_{j}\left(\theta_{jt}\right)}.$$

As the Markov chain is irreducible, aperiodic and Harris recurrent, the sample average converges to the expected value. That is,

$$T^{-1}\sum_{t=1}^{T}h\left(\theta_{jt}\right)\tilde{w}_{j}\left(\theta_{jt}\right)\xrightarrow{p}\int h\left(\theta\right)\tilde{w}_{j}\left(\theta\right)p\left(\theta\left|Y_{j}\right.\right)d\theta=\frac{p\left(Y\right)\int h\left(\theta\right)p\left(\theta\left|Y\right.\right)d\theta}{p\left(Y_{j}\right)k},$$

$$T^{-1}\sum_{t=1}^{T}\tilde{w}_{j}\left(\theta_{jt}\right)\xrightarrow{p}\int\tilde{w}_{j}\left(\theta\right)p\left(\theta\left|Y_{j}\right.\right)d\theta=\frac{p\left(Y\right)}{p\left(Y_{j}\right)k}.$$

It follows that $\sum_{t=1}^{T}h\left(\theta_{jt}\right)w_{j}\left(\theta_{jt}\right)\xrightarrow{p}\int h\left(\theta\right)p\left(\theta\left|Y\right.\right)d\theta$. □

3 Resample Move by Matched Samples

The conventional wisdom is that the importance sampling is doomed to degeneracy if the subposterior is substantially different from the full posterior. In this section, we consider a particle-jitter strategy similar to the resample-move method (Gilks and Berzuini, 2001). The idea is to use a Markov transition kernel to restore particle diversity. The simplest Markov kernel used to jitter particles is a random walk kernel. However, it is computationally intensive to evaluate the posterior densities when a particle randomly walks towards a new point that is not evaluated by the subposterior MCMC samplers. Our strategy is to restrict particles moving within the set of global proposals $\left\{\theta_{(1)}^{*},\ldots,\theta_{(N)}^{*}\right\}$, so that the previously evaluated posterior densities can be recycled.

Consider an extension of Algorithm 3 where steps 1 to 5 remain the same. If the importance weights are significantly unbalanced, we resample particles. Otherwise, we may skip resampling. To move particles, we specify a transition kernel $\tilde{q}\left(\theta^{*}\left|\theta\right.\right)$ and add the step 6 to Algorithm 3:

6 Move particles from $\theta_{jt0}\equiv\theta_{jt}$ to θ_{jts}, $s=1,\ldots,S$ by iterations:

6.1 Propose θ^* from $\left\{\theta^*_{(1)}, \ldots, \theta^*_{(N)}\right\}$ by rejection sampling with the target $\tilde{q}(\cdot)$.

6.2 Compute $\alpha = \min\left[1, \frac{p(\theta^*)\prod_{j=1}^m p(Y_j|\theta^*)\tilde{q}(\theta|\theta^*)}{p(\theta)\prod_{j=1}^m p(Y_j|\theta)\tilde{q}(\theta^*|\theta)}\right]$, where $\theta = \theta_{jt,s-1}$.

6.3 Set $\theta_{jts} = \begin{cases} \theta^* & \textit{with probability } \alpha \\ \theta & \textit{with probability } 1-\alpha \end{cases}$.

As is shown in Proposition 2, the posterior expectations can be approximated by the weighted average of the moved particles. Note that the weights are still evaluated at the original particles. That is, we use $\sum_{t=1}^T h(\theta_{jtS}) w_j(\theta_{jt})$ instead of $\sum_{t=1}^T h(\theta_{jtS}) w_j(\theta_{jtS})$.

Like the particle filter for static models (Chopin, 2002), it is possible to sequentially reweight moved particles if the invariant distribution of the transition kernel is the partial product of subposteriors $p(\theta)^{J/m}\prod_{j=1}^J p(Y_j|\theta), J \leq m$ to save computing costs.

Proposition 2. *Consider the extended Algorithm 3 with the step 6. Suppose that the MCMC samples are generated by irreducible, aperiodic and Harris recurrent Markov chains. As $T \to \infty$, we have*

$$\sum_{t=1}^T h(\theta_{jtS}) w_j(\theta_{jt}) \xrightarrow{p} \int h(\theta) p(\theta|Y) d\theta.$$

Proof. Rejection sampling in step 6.1 produces a draw θ^* asymptotically from $\tilde{q}(\theta^*|\theta)$, where $\theta = \theta_{jt,s-1}$. Steps 6.2 and 6.3 construct a transition kernel $r(\cdot)$ that has $p(\theta|Y)$ as its invariant distribution:

$$\int p(\theta|Y) r(\theta^*|\theta) d\theta = p(\theta^*|Y).$$

The weighted sample average is

$$\sum_{t=1}^T h(\theta_{jtS}) w_j(\theta_{jt}) = \frac{T^{-1}\sum_{t=1}^T h(\theta_{jtS})\tilde{w}_j(\theta_{jt})}{T^{-1}\sum_{t=1}^T \tilde{w}_j(\theta_{jt})}.$$

As in Proposition 1, we still have $T^{-1}\sum_{t=1}^T \tilde{w}_j(\theta_{jt}) \xrightarrow{p} \frac{p(Y)}{p(Y_j)k}$. Similarly, the sample average converges to the expected value:

$$T^{-1}\sum_{t=1}^T h(\theta_{jtS})\tilde{w}_j(\theta_{jt}) \xrightarrow{p} E\left[h(\theta_{jtS})\tilde{w}(\theta_{jt})\right].$$

As $T \to \infty$, the MCMC chain converges, with the marginal distribution of θ_{jt} being $p(\theta_{jt}|Y_j)$. The joint distribution of $\theta_{jt}, \theta_{jt1}, \ldots, \theta_{jtS}$ is $p(\theta_{jt}|Y_j)\prod_{s=1}^S r(\theta_{jts}|\theta_{jt,s-1})$, so we have

$E\left[h(\theta_{jtS})\tilde{w}_j(\theta_{jt})\right]$

$$= \int\int h\left(\theta^*\right) \tilde{w}_j\left(\theta\right) p\left(\theta\,|Y_j\,\right) r\left(\theta^*\,|\theta\,\right) d\theta d\theta^*$$

$$= \frac{p(Y)}{p(Y_j)k} \int\int h\left(\theta^*\right) p\left(\theta\,|Y\,\right) r\left(\theta^*\,|\theta\,\right) d\theta d\theta^*$$

$$= \frac{p(Y)}{p(Y_j)k} \int h\left(\theta^*\right) p\left(\theta^*\,|Y\,\right) d\theta^*$$

It follows that $\sum_{t=1}^{T} h\left(\theta_{jtS}\right) w_j\left(\theta_{jt}\right) \xrightarrow{p} \int h\left(\theta\right) p\left(\theta\,|Y\,\right) d\theta$. □

4 Numeric Examples

In this section, we assess the computational efficiency of importance sampling and resample-move methods with matched samples in comparison with the consensus Monte Carlo (CMC) from Scott et al. (2016). We address implementation issues on the choice of global and local proposals, as well as the number of samples (N and T).

We illustrate different global-local proposal specifications by three numerical examples. First, we design a larger-variance global proposal and smaller-variance local proposals located near subposterior means in Section 4.1. It is possible to calculate the upper bound of the local-global density ratio by Equation (3), which determines the ratio of N and T. Second, we design a global proposal identical to local proposal distributions, which implies that $N = T$ and subposterior densities used by MCMC samplers are fully recycled for evaluating the importance weights in Section 4.2. Third, we design a global proposal shared by subposterior random-walk MH samplers in Section 4.3. Local proposals are centered at the current state of MCMC chains. The upper bound of the local-global density ratio is updated with each parameter update.

4.1 Non-overlapping Beta Posteriors

We consider Beta-distributed subposteriors based on binomial data similar to those in Scott (2017). The first subset data have 90 successes out of 100 trials, while the second have 10 successes out of 110 trials. The parameter θ represents the probability of success. Under the uniform prior $Beta\left(1,1\right)$, the posterior and subposteriors in Equation (1) takes the form:

$$p\left(\theta\,|Y\,\right) \propto \theta^{a-1}\left(1-\theta\right)^{b-1},$$

$$p\left(\theta\,|Y_j\,\right) \propto \theta^{a_j-1}\left(1-\theta\right)^{b_j-1},$$

where $a = 101$, $b = 111$, $a_1 = 91$, $b_1 = 11$, $a_2 = 11$, $b_2 = 101$. That is, the two subposteriors are $Beta\left(a_1, b_1\right)$ and $Beta\left(a_2, b_2\right)$, which are nearly disjoint and distinct from the full posterior $Beta\left(a, b\right)$, as is shown in Figure 1.

To illustrate the general workflow of the divide-and-conquer methods, we run the MH samplers targeting the subposterior densities, instead of direct sampling from Beta distributions. The global proposals are generated from the Gaussian distribution $N\left(0.5, 0.3^2\right)$, while the local proposal distributions for the two subposteriors are specified as $N\left(0.7, 0.2^2\right)$ and $N\left(0.3, 0.2^2\right)$, respectively. By Equation (3), the upper bound of the local-global density ratio is around 2.2, so the number of global proposals and

MCMC samples satisfy $N \approx 2.2T$. About 10% of the local proposals are accepted by the subposterior MH samplers, but the rejected local proposals are not wasted, as the subposterior densities evaluated at the local proposals are recycled for evaluating the importance weights. To design efficient global and local proposal distributions, we strike a balance between the acceptance rates of the MH and rejection sampling. It is possible to increase the acceptance rates of the MH samplers by shifting the local proposals towards the subposterior means, say $N\left(0.9, 0.1^2\right)$ and $N\left(0.1, 0.1^2\right)$, but it needs more global proposals to support subposterior sampling ($N \approx 8T$).

Given the subposterior samples, we compare CMC and the importance sampling with and without resample-move steps.

CMC combines subposterior samples in a way that would be exact if the subposteriors were Gaussian. However, we have Beta-distributed subposteriors. Figure 1 indicates that CMC underestimates the full posterior mean and the standard deviation (std). The mean (std) estimator is 0.460 (0.021), while the true value is 0.476 (0.034) calculated from $Beta\left(a, b\right)$. The discrepancy does not decrease as $T \to \infty$, since inconsistency is rooted in the difference between Beta and Gaussian posterior distributions.

Table 1 shows a sequence of estimators produced by our method. The second column provides the subposterior mean and std, which are consistent with the moments of $Beta\left(a_1, b_1\right)$ and $Beta\left(a_2, b_2\right)$. The third column shows the estimators by the importance sampling. In theory, the weighted particles approximate the full posterior asymptotically. The non-overlapping subposteriors cause weight concentration on small particles of the first subset, and large particles of the second subset, hence a poor approximation. However, the resample-move method rejuvenates particles. After the first iteration ($S = 1$), we obtain two greatly improved estimators of the full posterior mean (std): 0.582 (0.170) and 0.382 (0.161). The following iterations witness continuous improvement of estimators, which quickly converge to the true posterior mean (std) 0.476 (0.034). In this example, 25 iterations suffice to produce an accurate estimator.

In summary, CMC is computationally attractive, but does not necessarily produce consistent estimators under non-Gaussian distributions. The importance sampling estimators are consistent, but the approximation could be crude due to weight degeneracy. The resample-move steps provide accurate estimates of the full posterior, but the computing time is longer. It takes approximately 5 seconds to process 50000 subposterior samples on a laptop computer, while CMC can combine samples in less than 0.1 seconds.

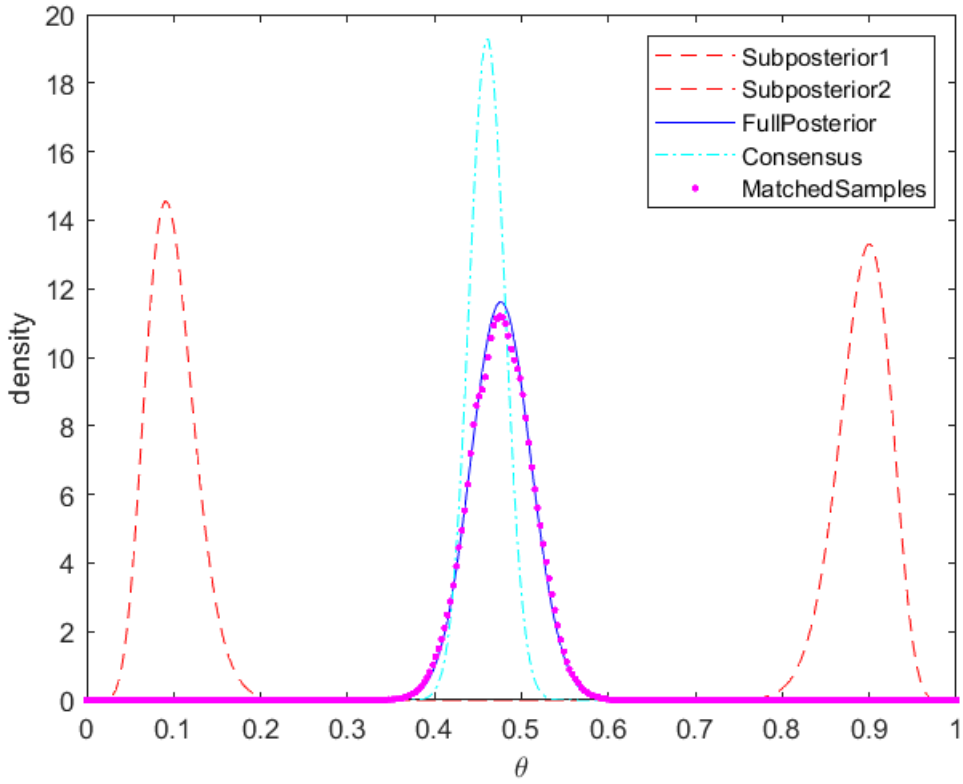

Figure 1: Non-overlapping Beta posteriors. The dashed lines plot two subposteriors. The dash-dot and dotted lines plot the full posterior estimates by the consensus Monte Carlo and our algorithm, respectively. The sold line is the ground truth calculated from $Beta\left(a,b\right)$.

	$\sum h\left(\theta\right)/T$	$\sum h\left(\theta\right)w\left(\theta\right)$	$S=1$	$S=10$	$S=25$	Truth	Consensus
Subset1	0.892	0.743	0.582	0.477	0.477	0.476	0.460
	(0.031)	(0.003)	(0.170)	(0.051)	(0.035)	(0.034)	(0.021)
Subset2	0.098	0.236	0.382	0.477	0.477	0.476	0.460
	(0.028)	(0.003)	(0.161)	(0.052)	(0.036)	(0.034)	(0.021)

Table 1: Non-overlapping Beta posterior mean and std (in parenthesis) estimators. The second column shows the subposterior mean and std. The third column shows the importance sampling estimators in step 5 of Algorithm 3. The next three columns show the resample-move estimators, with the number of iterations $S=1,10,25$. The *Truth* column provides the ground truth calculated from $Beta\left(a,b\right)$. The last column shows the consensus Monte Carlo results.

4.2 Logistic Regression with Unidentified Parameters

Scott et al. (2016) and Nemeth and Sherlock (2018) consider a logistic regression with five binary predictors, of which the last one (x_5) is a rarely occurring variable (with the probability 0.01) that is highly predictive of an event if it occurs. The data of Scott et al. (2016) have 10000 observations, which are randomly partitioned into $m = 100$ subsets. The probability that $x_5 = 0$ for all data points in a subset is $0.99^{100} = 0.366$, under the assumption of random sampling.

We specify a standard Gaussian prior $p(\theta) \propto e^{-\frac{\theta'\theta}{2}}$ for the predictor coefficients, so that the subset prior is $p(\theta)^{1/m} \propto e^{-\frac{\theta'\theta}{2m}}$, which preserves normality with the variance m. If $x_5 = 0$ for all data points in a subset, the corresponding coefficient θ_5 has no impact on the likelihood function, hence an unidentified parameter whose subposterior is identical to the prior. In contrast, some subsets have multiple occurrences of $x_5 = 1$, and the subposterior mode of θ_5 may be remote from zero, as x_5 is highly predictive.

As we have 100 subposteriors, it is tedious to manually specify a proposal distribution for each subposterior MH sampler. Our strategy is numerical optimization of the subposterior density by 10% of the full data. The mean and variance of the proposal distribution are specified as the maximum-a-posteriori (MAP) estimator and the inverse Hessian evaluated at MAP, respectively. Due to heterogeneity of θ_5 such that an unidentified parameter has the mean 0 and an identified parameter has the mean around 3, we increase the proposal variance of θ_5 to 3. That is, both global and local proposals are generated from the Gaussian distribution with the mean

$$\mu = \begin{pmatrix} -2.842 & 1.192 & -0.355 & 0.606 & 2.416 \end{pmatrix}'$$

and the covariance matrix

$$\Sigma = \begin{pmatrix} 0.040 & -0.018 & -0.014 & -0.029 & -0.012 \\ -0.018 & 0.048 & -0.001 & 0.001 & 0.005 \\ -0.014 & -0.001 & 0.059 & -0.001 & -0.006 \\ -0.029 & 0.001 & -0.001 & 0.047 & 0.005 \\ -0.012 & 0.005 & -0.006 & 0.005 & 3.000 \end{pmatrix}.$$

Figure 2 provides the kernel density of an unidentified subposterior of θ_5, which is nearly flat. It also shows the kernel density estimates of the full posterior by CMC and importance sampling with matched samples (Algorithm 3). As a benchmark, we run Algorithm 1 on the full data to obtain a ground truth of the full posterior density. The full posterior mean of θ_5 is 3.27. Both CMC and our algorithm successfully approximate the posterior mean by combining subposterior samples. However, CMC over-estimates the posterior variance, as the dash-dot line is flatter than the solid line in Figure 2. The results echo Figure 7(a) in Scott et al. (2016), where their matrix-weighting kernel density plot (dashed line) is flatter than the overall density plot (solid line). Figure 2 shows that our algorithm (dotted line) provides a consistent estimator of the full posterior density, and the estimated variance is accurate. Of course, CMC remains to be computationally attractive, as it only takes one second to combine subposterior samples, while our algorithm costs about 25 seconds to weigh subposterior samples.

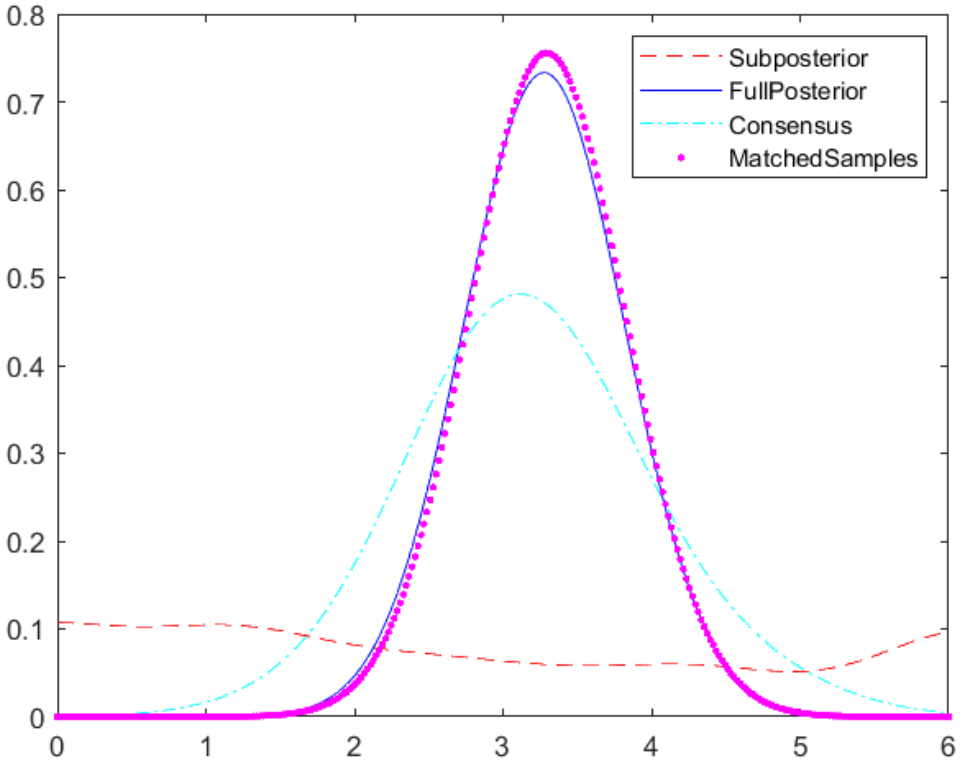

Figure 2: Logistic regression with an unidentified parameter. The dashed line corresponds to the subposterior density, while other lines provide the full posterior density estimators of θ_5. The solid line serves as the ground truth obtained by full-data MCMC simulation. The dash-dot line shows the consensus Monte Carlo results, and the dotted line corresponds to the importance sampling by matched samples (Algorithm 3).

Figure 2 indicates that the current proposal leads to a good approximation of the full posterior distribution, but it is possible to refine the proposal specification for better approximations. A common proposal density cannot be close to all of the 100 subposterior densities, and it is tempting to customize a local proposal $q_j(\cdot) = \phi(\cdot; \mu_j, \Sigma_j)$ for each subposterior. Instead of manual tuning for each subposterior sampler, we consider a trial-and-error tuning approach. First, we use the current proposal $q_j(\cdot) = q(\cdot) = \phi(\cdot; \mu, \Sigma)$ for an initial estimate of the subposterior means. Second, we tune local proposals by specifying μ_j as a weighted average of the subposterior means and μ. Also, we specify $\Sigma_j < \Sigma$. For example, $\Sigma_j = \Sigma/2$. The computing cost of the sampler tuning is moderate, as global proposal draws can be reused and subposterior densities can be recycled.

4.3 Multimodal Gaussian Mixture Distribution

Nemeth and Sherlock (2018) consider a mixture of two bivariate Gaussian distributions

$$p(Y_i \,|\theta_1, \theta_2) = \frac{1}{2}\phi\left(Y_i; \theta_1\iota, \sigma^2 I\right) + \frac{1}{2}\phi\left(Y_i; \theta_2\iota, \sigma^2 I\right),$$

where $\phi(\cdot)$ is the Gaussian density, ι is a vector of ones and I is an identity matrix. Observations are generated by the true parameters $\theta_1 = 0.1$ and $\theta_2 = -0.1$.

Let the prior be $p(\theta_1, \theta_2) = \phi(\theta_1, \theta_2; 0\iota, 100I)$. Because the prior and the likelihood are invariant to switching labels of θ_1 and θ_2, the posterior density is likely to exhibit two modes near $(0.1, -0.1)$ and $(-0.1, 0.1)$.

With a mesh-grid of (θ_1, θ_2) points, the left panels of Figure 3 provide contour plots of the full posterior density (up to a normalizing constant) under three scenarios: $\sigma^2 = \{0.5, 1, 2\}$. As σ^2 decreases, the high-density regions concentrate around the two modes. For the first scenario, the supports of the posterior are nearly disjoint. The second scenario has connected supports with two distinct modes. The posterior in the third scenario appears unimodal.

We evaluate the performance of Algorithm 3 in terms of approximating the multimodal posterior. Applications are not limited to the independence MH samplers, as our algorithm is also applicable to the random walk MH samplers with matched samples. In this example, the global proposal distribution is specified as a bivariate normal distribution with the mean $(0, 0)$ and the covariance matrix $\Sigma = \begin{pmatrix} 0.08 & -0.07 \\ -0.07 & 0.08 \end{pmatrix}$. The negative correlation indicates that a positive θ_1 draw is likely to be paired by a negative θ_2 draw. The local proposal distribution is specified as a bivariate normal distribution with the covariance matrix $\Sigma/2$ and the mean at the current state of the random-walk MH sampler. By Equation (3), the upper bound of the local-global density ratio is a function of (θ_1, θ_2), and thus determined at each iteration of the MCMC sampling. In the case of $\sigma^2 = 0.5$, it takes about 2.6 iterations to generate local proposals using rejection sampling (that is, $N \approx 2.6T$). The acceptance rate of the MH sampler is around 19.4%.

The middle panels of Figure 3 show the contour plots of the kernel density estimates of the full posterior by Algorithm 3. The results indicate that our method is able to

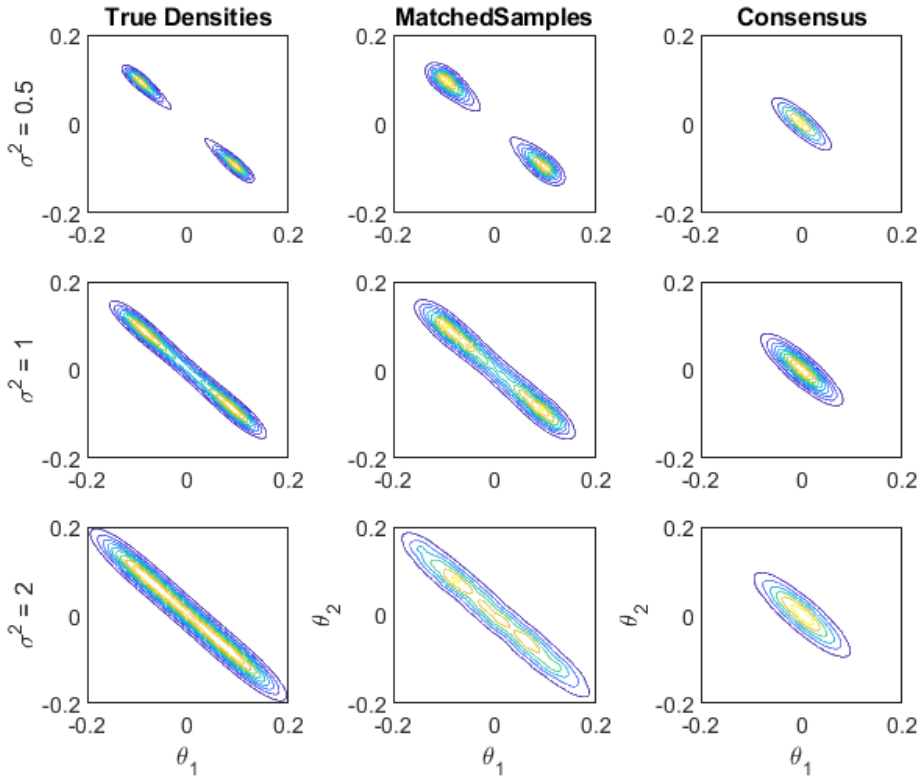

Figure 3: Posterior contour plots of multimodal Gaussian mixture densities. The left panels are the true densities evaluated by a grid of (θ_1, θ_2) points. The middle panels are kernel density estimates by Algorithm 3. The right panels are the results of the consensus Monte Carlo. The three rows correspond to the scenarios of $\sigma^2 = \{0.5, 1, 2\}$.

identify the posterior modes in all scenarios, and the overall shapes of the density estimates are close to the true densities. In comparison, the right panels of Figure 3 show the results of CMC, which are always unimodal and not designed for approximating a multimodal posterior.

5 An Application

Many divide-and-conquer MCMC methods assume data partition at random, which not only increases the cost of data access, but also loses economic interpretation of the subposteriors. It is of interest to study subset data sorted by years, entities or some specific features.

In this application, we consider a real-world mortgage dataset, in which each subset contains loans originated from a specific lender. The appeal of lender-specific data partition is that every bank has its own loan decision rules. A home buyer is reluctant to do credit shopping at multiple banks for the maximum chance of loan approval and the best rate, since hard inquiries stain the credit score of the applicant. Big data help home buyers estimate the mortgage approval probabilities and rates from different banks, although this is not the focus of the paper. The aim of this section is to illustrate the divide-and-conquer logistic regressions and evaluate the performance of our method in the presence of heterogeneous subposteriors under non-random data partition.

The U.S. Home Mortgage Disclosure Act (HMDA) requires financial institutions to disclose mortgage data to the public. In the year 2018 alone, over 5600 institutions reported 15 million loans to HMDA database. We extract the data of 10 largest lenders in terms of the number of loans. Each subset contains the data of a lender. The subset sample sizes are 0.97, 0.55, 0.51, 0.46, 0.37, 0.24, 0.24, 0.22, 0.20, 0.18 million, totaling 3.94 million observations. We run lender-specific logistic regressions to predict whether a loan is originated or denied, using the predictors of the debt-to-income ratio (DIR), loan size, loan purpose, applicant gender, race and ethnicity, as well as an indicator of joint filing.

Similar to global and local proposals specified in Section 4.2, we first obtain subset MAP and Hessian by numerical optimization with 0.5% of data, and then set MAP as the mean of global and local proposal distributions, and the covariance matrix is constructed based on the inverse Hessian. With fully matched samples ($N = T$), all subposterior densities are recycled for importance sampling and resample-move steps.

The lender-by-lender subposteriors are summarized in the upper part of Table 2. The subposterior means vary substantially from bank to bank, but the sign is unambiguous with an economic interpretation: 1) a low DIR is crucial to mortgage application; 2) a larger-size loan increases the chance of approval; 3) a loan for home purchase is more likely to be approved, compared to refinancing and home improvement; 4) female applicants are considered favorably; and 5) joint filing increases the probability of loan approval.

For each subposterior MCMC chain, the subposterior samples are weighted, resampled and moved, so we have 10 estimators shown in the bottom part of Table 2. Because all of them are consistent estimators of the full posterior, they should validate each other. Table 2 shows that most estimators are highly close to each other. Regarding the key factor DIR, 9 estimators indicate that the effect is around -5.4, but one of them (lender 3) provides a slightly different value of -5.5. Figure 4 plots the sequence of the full-posterior DIR estimators as the number of iterations (S) increases. It demonstrates that 9 out of 10 estimators converge quickly in a few iterations ($S < 20$).

	DIR	loanSize	purpose	male	white	hispanic	couple
Subposteriors:							
Lender 1	-7.081	2.438	1.512	-0.295	0.239	-0.482	0.060
Lender 2	-4.348	2.282	0.369	-0.294	0.091	-0.399	-0.011
Lender 3	-6.569	2.435	1.542	-0.361	0.235	-0.463	0.012
Lender 4	-4.763	1.695	0.555	-0.329	0.224	-0.558	0.025
Lender 5	-4.84	1.708	1.248	-0.291	0.211	-0.496	0.085
Lender 6	-3.696	1.664	1.933	-0.371	0.103	-0.409	-0.022
Lender 7	-4.657	1.324	2.191	-0.276	0.239	-0.461	0.078
Lender 8	-3.351	1.104	1.517	-0.274	0.164	-0.438	0.037
Lender 9	-4.038	2.179	1.118	-0.294	0.104	-0.447	-0.074
Lender 10	-7.441	4.072	0.718	-0.317	0.112	-0.433	-0.052
Full Posteriors:							
Lender 1	-5.396	2.187	1.226	-0.292	0.162	-0.462	0.006
Lender 2	-5.395	2.186	1.224	-0.292	0.163	-0.462	0.006
Lender 3	-5.509	2.211	1.265	-0.306	0.167	-0.469	0.007
Lender 4	-5.388	2.182	1.228	-0.292	0.163	-0.462	0.006
Lender 5	-5.369	2.167	1.227	-0.29	0.164	-0.463	0.009
Lender 6	-5.387	2.182	1.229	-0.292	0.162	-0.461	0.007
Lender 7	-5.397	2.184	1.23	-0.291	0.162	-0.461	0.006
Lender 8	-5.399	2.187	1.226	-0.291	0.162	-0.462	0.007
Lender 9	-5.400	2.187	1.227	-0.292	0.163	-0.461	0.007
Lender 10	-5.397	2.185	1.226	-0.292	0.163	-0.461	0.006

Table 2: Lender-specific logistic regressions. For each subposterior MCMC chain, the subposteriors means are reported. Then, particles are weighted, resampled and moved, which produces 10 estimators of the full posterior.

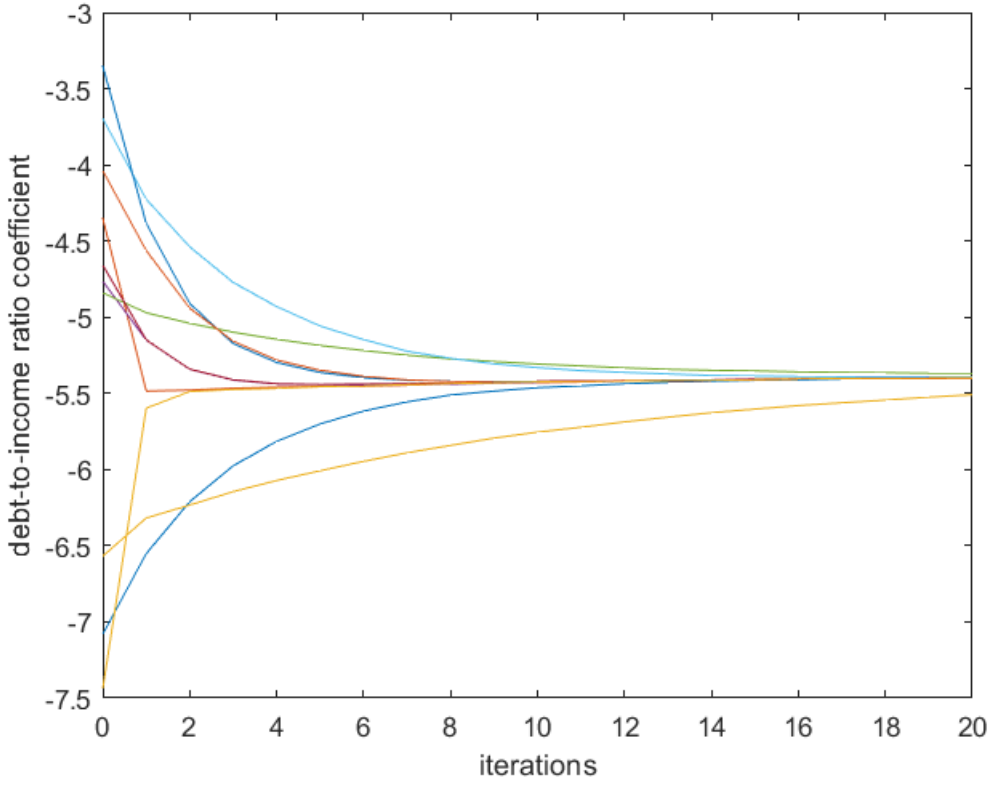

Figure 4: Posterior mean estimators of DIR coefficient. Lender-specific logistic regressions produce 10 subposterior MCMC chains. For each chain, particles are weighted, resampled and moved, which produces 10 estimator sequences. Iteration 0 corresponds to the subposterior mean of the DIR coefficient. The following iterations produce the full posterior mean estimators.

6 Discussion

As a combination of rejection sampling, MCMC simulation and importance weighting, Algorithm 3 provides an asymptotically exact divide-and-conquer method for Bayesian inference with massive data. The strengths of the algorithm include 1) efficiency, as the densities used by the subposterior MH samplers are recycled for the importance weight evaluation, 2) degeneracy rescue, as the recycled subposterior densities facilitate the resample-move method for promoting diversity of particles, and 3) convenience of coding, as the global proposals can be created by using the same random seed across all subposterior simulations.

We conclude the paper by discussing two open questions of implementing the algorithm: how to specify a reasonable proposal density, and how to select or average estimators.

First, the proposal distribution of the MH sampler needs careful tailoring to the underlying posterior to work well. For moderate-size data, a common strategy is to use numerical optimization to maximize the log posterior and calculate the Hessian at the posterior mode. A Gaussian or t proposal distribution centers at the posterior mode and has the covariance matrix proportional to the inverse Hessian. See Smets and Wouters (2007) for an econometric application. It is understood that numerical optimization is computationally intensive and the Hessian could be poorly approximated. To adapt the strategy to massive data scenarios with moderate computing cost, we consider revising the proposal specification after a trial of Algorithm 3 under $q_j\left(\theta^* \left|\theta\right.\right) = q\left(\theta^*\right)$. With the matched samples, we are able to use the recycled densities to evaluate the log posterior. Assume that the log posterior is approximated by a quadratic function: $\ln p\left(\theta \left| Y\right.\right) = c + \theta'\beta + \frac{1}{2}\theta' H\theta$, where the coefficients c, β, H are estimated by least squares using the "regression data": $\theta^*_{(n)}$ and $\ln p\left(\theta^*_{(n)} \left| Y\right.\right)$. A reasonable approach is to specify the Gaussian proposal with the mean $-H^{-1}\beta$ (the approximated posterior mode) and the covariance matrix proportional to $-H^{-1}$. It may not always be a good proposal, but suggests a way to recycle the subposterior densities to estimate the location and scale of the posterior with minimum passes of data.

Second, a feature of Algorithm 3 is that m estimators are produced simultaneously. That is, for each and every subposterior chain, the weighted particles consistently estimate the full posterior expectations. The feature provides a self-validation instrument in that estimators should be close to each other. However, selection or averaging the m consistent estimators remains an open question. Any linear combination of the estimators preserves consistency. It is tempting to take the simple average, but it is not necessarily the optimal strategy. For example, visual inspection of Figure 4 suggests that 9 out of 10 estimators are nearly identical after a few resample-move iterations, but there is an outlier that converges slower than others. Heuristically, we may discard or down-weight the outlier, but automatic selection or averaging requires future work on the analysis of the estimator variances and covariances, which can be an interesting and challenging task, since the matched samples are used in the subposterior MCMC chains.

References

Banterle, M., C. Crazian, A. Lee, and C. Robert (2019). Accelerating Metropolis-Hastings algorithms by delayed acceptance. *Foundations of Data Science 1*, 103–128. 2

Bardenet, R., A. Doucet, and C. Holmes (2017). On Markov Chain Monte Carlo methods for tall data. *Journal of Machine Learning Research 18*, 1–43. 1, 2

Bierkens, J., P. Fearnhead, and G. Roberts (2019). The zig-zag process and super-efficient sampling for Bayesian analysis of big data. *The Annals of Statistics 47*, 1288–1320. 2

Chopin, N. (2002). A sequential particle filter method for static models. *Biometrika 89*, 539–551. 8

Gilks, W. R. and C. Berzuini (2001). Following a moving target - Monte Carlo inference for dynamic Bayesian models. *Journal of the Royal Statistical Society. Series B 63*, 127–146. 2, 7

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

Korattikara, A., Y. Chen, and M. Welling (2014). Austerity in MCMC land: Cutting the Metropolis-Hastings budget. *Proceedings of the 31st International Conference on Machine Learning 1*, 181–189. 2

Maclaurin, D. and R. Adams (2014). Firefly Monte Carlo: Exact MCMC with subsets of data. *Uncertainty in Artificial Intelligence - Proceedings of the 30th Conference, UAI 2014*, 4289–4295. 2

Metropolis, N., A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics 21*, 1087–1092. 1

Minsker, S., S. Srivastava, L. Lin, and D. Dunson (2017). Robust and scalable Bayes via a median of subset posterior measures. *Journal of Machine Learning Research 18*, 1–40. 1

Neiswanger, W., C. Wang, and E. P. Xing (2014). Asymptotically exact, embarrassingly parallel MCMC. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, pp. 623–632. Arlington: AUAI Press. 1

Nemeth, C. and C. Sherlock (2018). Merging MCMC subposteriors through Gaussian-process approximations. *Bayesian Analysis 13*(2), 507–530. 2, 12, 14

Quiroz, M., R. Kohn, M. Villani, and M. Tran (2019). Speeding up mcmc by efficient data subsampling. *Journal of the American Statistical Association 114*, 831–843. 2

Scott, S. L. (2017). Comparing consensus Monte Carlo strategies for distributed Bayesian computation. *Brazilian Journal of Probability and Statistics 31*, 668–685. 3, 9

Scott, S. L., A. W. Blocker, F. V. Bonassi, H. A. Chipman, E. I. George, and R. E. McCulloch (2016). Bayes and big data: The consensus Monte Carlo algorithm. *International Journal of Management Science and Engineering Management 11*, 78–88. 1, 9, 12

Smets, F. and R. Wouters (2007). Shocks and frictions in US business cycles: A Bayesian DSGE approach. *American Economic Review 97*, 586–608. 19

Srivastava, S., C. Li, and D. Dunson (2018). Scalable Bayes via barycenter in wasserstein space. *Journal of Machine Learning Research 19*, 1–35. 1

Teh, Y. W., H. A. Thiery, and S. J. Vollmer (2016). Consistency and fluctuations for stochastic gradient langevin dynamics. *Journal of Machine Learning Research 17*, 1–33. 2

Wang, X. and D. Dunson (2013). Parallelizing mcmc via weierstrass sampler. *arXiv preprint arXiv:1312.4605*. 1

Wang, X., F. Guo, K. Heller, and D. Dunson (2015). Parallelizing mcmc with random partition trees. *Advances in Neural Information Processing Systems*, 451–459. 1