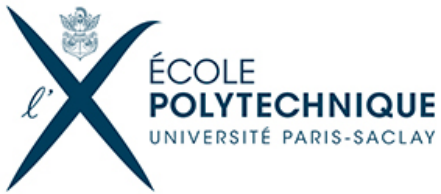


On Markov chain Monte Carlo methods for tall data by Bardenet R., Doucet A., Holmes C. (2017).

ADEIKALAM Pierre
CHEN Guangyue
MORALES Katherine
XU Kevin

The paper

Aim of the paper

The authors tackle the problem of applying Markov chain Monte Carlo (MCMC) methods for Bayesian inference on **tall** datasets where the number of observations n is **very large**.

- Presentation of a comprehensive review of the existing literature, theoretical guarantees and the limitations of each method.
- Propose a promising approach based on a technique called confidence sampling.

Our aim

- Better understand why the naive approaches we would have naturally tried ourselves are not good ideas in their basic form.
- Present a promising approach to scaling up MCMC methods to tall datasets.

Contenu

Bayesian approach

- Let $p(\theta)$ be the prior distribution.
- Let $\pi(\theta)$ be the posterior distribution of θ :

$$\pi(\theta) = p(\theta|X) = \frac{p(\theta)\prod_{i=1}^{n} p(x_i|\theta)}{p(X)} \quad \propto \quad \gamma(\theta), \qquad (1)$$

where $\gamma(\theta) := p(\theta)e^{(n\, l(\theta))}$ is the unnormalized version of π and $l(\theta)$ is the associated average log-likelihood

$$l(\theta) = \frac{1}{n}\sum_{i=1}^{n} \log p(x_i|\theta). \qquad (2)$$

Metropolis Hastings Algorithm

Metropolis-Hastings (MH) algorithm simulates a Markov chain $(\theta_k)_{k\geq 1}$ of invariant distribution π,

Algorithm 1 : Metropolis-Hastings

Input : $\gamma(\cdot)$: unnormalized version of π
$q(\cdot|\cdot)$: proposal distribution
N_{iter} : number of iterations
Output: $(\theta_k)_{k=1,\dots,N_{iter}}$

for $i = 1, \dots, N_{iter}$ **do**

1. $\theta \leftarrow \theta_{k-1}$
2. $\theta' \sim q(.|\theta)$
3. $\alpha(\theta, \theta') \leftarrow \frac{\gamma(\theta')}{\gamma(\theta)} \frac{q(\theta|\theta')}{q(\theta'|\theta)}$
4. $U \sim \mathcal{U}(0,1)$
5. **If** $U < \alpha(\theta, \theta')$: $\theta_k \leftarrow \theta'$
6. **else :** $\theta_k \leftarrow \theta$

end

The observations of the dataset are independent. Thus, we can compute the acceptance ratio with :

$$\log \alpha(\theta, \theta') = \log \left[\frac{p(\theta')}{p(\theta)} \frac{q(\theta|\theta')}{q(\theta'|\theta)} \right] + n\big[l(\theta') - l(\theta)\big]. \qquad (3)$$

Problem

Since $nl(\theta) = \sum_{i=1}^{n} \log p(x_i|\theta)$, computing $n[l(\theta') - l(\theta)]$ requires a full pass over the dataset.

Approaches :

- Divide and conquer approaches
- Exact subsampling approaches
- Approximate subsampling approaches

Divide and conquer Methods

Divide-and-conquer

Divide and conquer approaches appear as a natural way of handling tall data as they divide the initial dataset into batches, run MCMC on each batch in parallel, and then combine these results to obtain an approximation of the posterior distribution.

Formally, assuming the data X is divided into batches $x_1, ..., x_B$, the latter task can be written as :

$$\pi(\theta) = p(\theta|X) \propto \prod_{i=1}^{B} p(\theta)^{\frac{1}{B}} p(x_i|\theta) \tag{4}$$

which is justified **only if we assume the batch posteriors π_i to be Gaussian.**

Divide-and-conquer

Two main problems arise :

1. **MCMC methods do not return posteriors as mathematical functions**, so an extra step is needed to combine the batch posteriors using this trick.
2. **How to efficiently combine non-Gaussian batch posteriors ?**

Divide-and-conquer

The authors mention different ways of combining the batch posteriors :

- Fit a Gaussian approximation to the MCMC draws of each batch posterior and multiply them. (Huang and Gelman, 2005)
- Compute a weighted average of the MCMC draws of each batch posterior. Under Gaussian assumptions , this average should follow the same distribution as the combined batch posteriors (Scott et al., 2013).
- For each batch x_i, target the artificial posterior $\pi_i(\theta) \propto p(\theta)^{\frac{1}{B}} p(x_i|\theta)$. Then, fit a smooth approximation to each batch posterior π_i and multiply them (Neiswanger et al., 2014).

Divide-and-conquer

However, these methods are theoretically justified only when the batch posteriors are Gaussian, or **when the size of each batch goes to infinity**, which defeats the purpose of this approach.

The supports of the π_i can be disjoint and as a result the product of their approximations will yield a poor approximation of π.

Exact Subsampling Approach : Pseudo-Marginal Metropolis-Hastings

Pseudo-Marginal MH

Pseudo-Marginal MH consists in using an unbiased estimator $\hat{\gamma}(\theta)$ of the unnormalized target distribution $\gamma(\theta)$ instead of evaluating $\gamma(\theta)$ at each iteration of the MH algorithm.

The acceptance ratio becomes :

$$\hat{\alpha}(\theta, \theta') = \frac{\hat{\gamma}(\theta')}{\hat{\gamma}(\theta)} \frac{q(\theta|\theta')}{q(\theta'|\theta)} \tag{4}$$

We hope that asymptotically, the MCMC has the target distribution we were aiming for.

Exact Subsampling Approach : Pseudo-Marginal Metropolis-Hastings

The simplest estimator for the loglikelihood we could build is :

$$n\hat{l}(\theta) = \frac{n}{t}\sum_{i=1}^{t} \log p(x_i^*|\theta)$$

which is an unbiased estimator of $nl(\theta)$.

Exact Subsampling Approach : Pseudo-Marginal Metropolis-Hastings

The problems begin when we realise that :

$$\mathbb{E}[e^{n\hat{l}(\theta)}] \neq e^{\mathbb{E}[n\hat{l}(\theta)]} \tag{5}$$

which means that $e^{n\hat{l}(\theta)}$ is not an unbiased estimator of $e^{nl(\theta)} = \gamma(\theta)$.
Therefore, the target distribution is not preserved and this estimator does not work for an exact subsampling approach.

Exact Subsampling Approach : Pseudo-Marginal Metropolis-Hastings

Just build an unbiased estimator of $e^{nl(\theta)}$ then !

Jacob and Thierry (2015) have shown that it is actually impossible to build an unbiased estimator of $e^{nl(\theta)}$ with just an unbiased estimator of $nl(\theta)$ without making further assumptions.

And even if we were to make further assumptions, building an unbiased estimator of $e^{nl(\theta)}$ is highly non-trivial and controlling its variance is extremely difficult.

Approximate Subsampling Approach : Naive Subsampling

Naive Subsampling

Assuming that we have given up on exactly targeting $\gamma(\theta)$, naive subsampling is a simple technique similar to Pseudo-Marginal MH that uses the unbiased estimator :

$$\hat{l}(\theta) = \frac{1}{n} \sum_{i=1}^{n} \frac{z_i}{\lambda} l_i(\theta)$$

where the $z_i \sim B(1, \lambda)$ are i.i.d.

Under Gaussian assumptions, Doucet et al. (2015) have shown that Pseudo-Marginal MH can still be effective if the variance of the loglikelihood estimation is kept near 1.

With this simple approach, the variance of loglikelihood estimation is easy to compute :

$$Var[n\hat{l}(\theta)] = \frac{(1-\lambda)}{\lambda} \sum_{i=1}^{n} l_i(\theta)^2$$

Problems :

- Since $\sum_{i=1}^{n} l_i(\theta)^2$ grows with n, for $Var[n\hat{l}(\theta)]$ to be close to 1 we need λ to be close to 1.
- If λ is close to 1 there is no significant advantage between using this technique or the complete data, which defeats the purpose of this approach.

Approximate Subsampling Approach : Confidence Sampler

Confidence Sampling

Instead of approximating $\gamma(\theta)$, the confidence sampler will directly try to approximate the acceptance ratio $\alpha(\theta, \theta')$.

If our approximation $\hat{\alpha}(\theta, \theta')$ is good enough, it should yield the same acceptance decision as $\alpha(\theta, \theta')$ and the stationary distribution should be preserved.

As with any approximation, there is no way to guarantee its quality with 100% confidence, but we can guarantee it with 95% confidence (for example).

Approximate Subsampling Approach : Confidence Sampler

For this we just need to assume that :

$$|\log \hat{\alpha}(\theta, \theta') - \log \alpha(\theta, \theta')| \leq C_{\theta,\theta'}$$

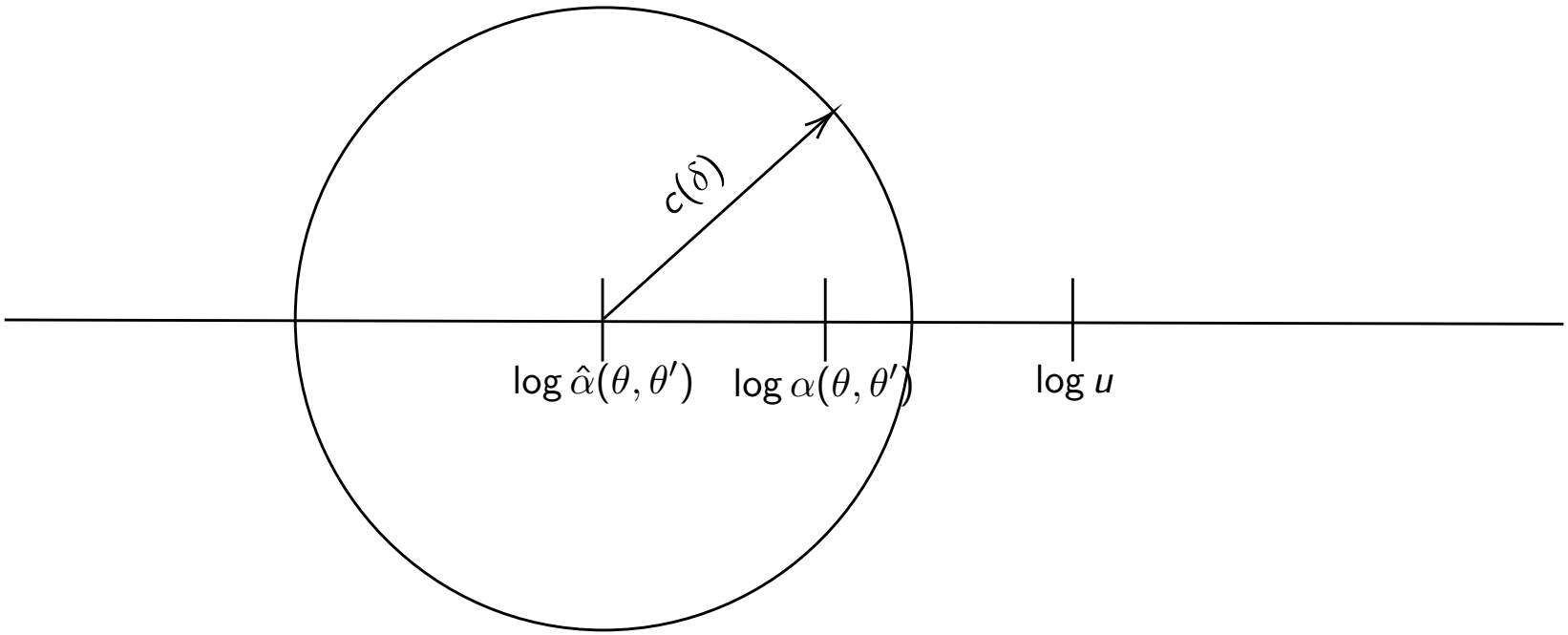
The fact that this difference is bounded is what allows us to use a concentration inequality.

$$P(|\log \hat{\alpha}(\theta, \theta') - \log \alpha(\theta, \theta')| \leq c(\delta)) \geq 1 - \delta$$

Approximate Subsampling Approach : Confidence Sampler

On the event $\{|\log \hat{\alpha}(\theta, \theta') - \log \alpha(\theta, \theta')| \leq c(\delta)\}$:

If we have that $|log\hat{\alpha}(\theta, \theta') - \log(u)| \geq c(\delta)$, then the decision yielded by $\hat{\alpha}(\theta, \theta')$ will be the same as the one yielded by $\alpha(\theta, \theta')$

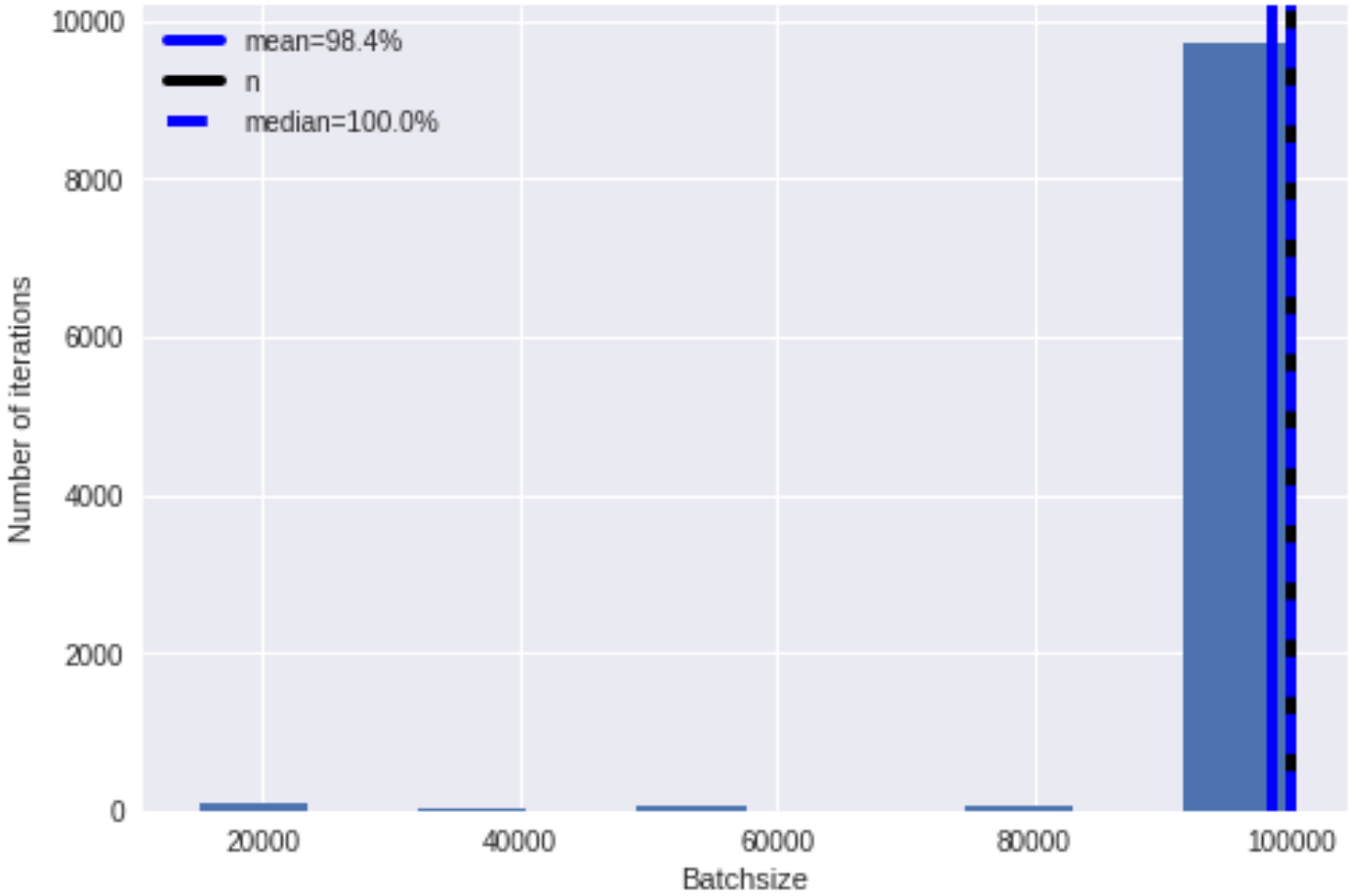


Approximate Subsampling Approach : Confidence Sampler

In practice what happens is that the approximation $\hat{\alpha}(\theta, \theta')$ depends on t samples and we sequentially increase t towards n until we are confident enough that $\log u$ is outside the confidence ball.

Now, if we try to implement this approach, we see that the number of likelihood evaluations is still close to n, so no significant improvement was made.

Example : one-dimensional normal distribution to 10^5 i.i.d. points drawn according to $\mathcal{N}(0, 1)$.



Improved confidence sampler

Proposition of the authors : a modified version of the confidence sampler

Use of a proxy to replace the approximated loglikelihood ratio
$\rightarrow$ **If we are *confident enough* in our proxy, the acceptance decision should be identical.**

The trick is that this new algorithm can require less than $\mathcal{O}(n)$ likelihood evaluations per iteration.

Improved confidence sampler

Assumptions on the proxies

The conditions required for this algorithm to work are the following : For $\theta, \theta' \in \Theta$

1. $\wp_i(\theta, \theta') \approx l_i(\theta') - l_i(\theta)$ (the proxy approximates the loglikelihood ratio)
2. $\sum_{i=1}^{n} \wp_i(\theta, \theta')$ is cheap to compute
3. $|l_i(\theta') - l_i(\theta) - \wp_i(\theta, \theta')|$ can be bounded uniformly for $i \in [\![1, n]\!]$ and the bound is cheap to compute.

The third condition allows us to use a concentration inequality and bound the confidence in the approximated acceptance ratio.

Example Proxy : Taylor Expansion

The log likelihood around some reference value θ_* can be approximated by :

$$\hat{l}_i(\theta) = l_i(\theta_*) + g_{i,*}^T(\theta - \theta_*) + \frac{1}{2}(\theta - \theta_*)^T H_{i,*}(\theta - \theta_*)$$

where $g_{i,*}$ is the gradient of l_i at θ_* and $H_{i,*}$ is the Hessian matrix of l_i at θ_*.

The proxy can then be defined by :

$$\wp_i(\theta, \theta') = \hat{l}_i(\theta') - \hat{l}_i(\theta) \approx l_i(\theta') - l_i(\theta)$$

Example Proxy : Taylor Expansion

If we have already precomputed :

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} g_{i,*}$$

$$\hat{S} = \frac{1}{n} \sum_{i=1}^{n} H_{i,*}$$

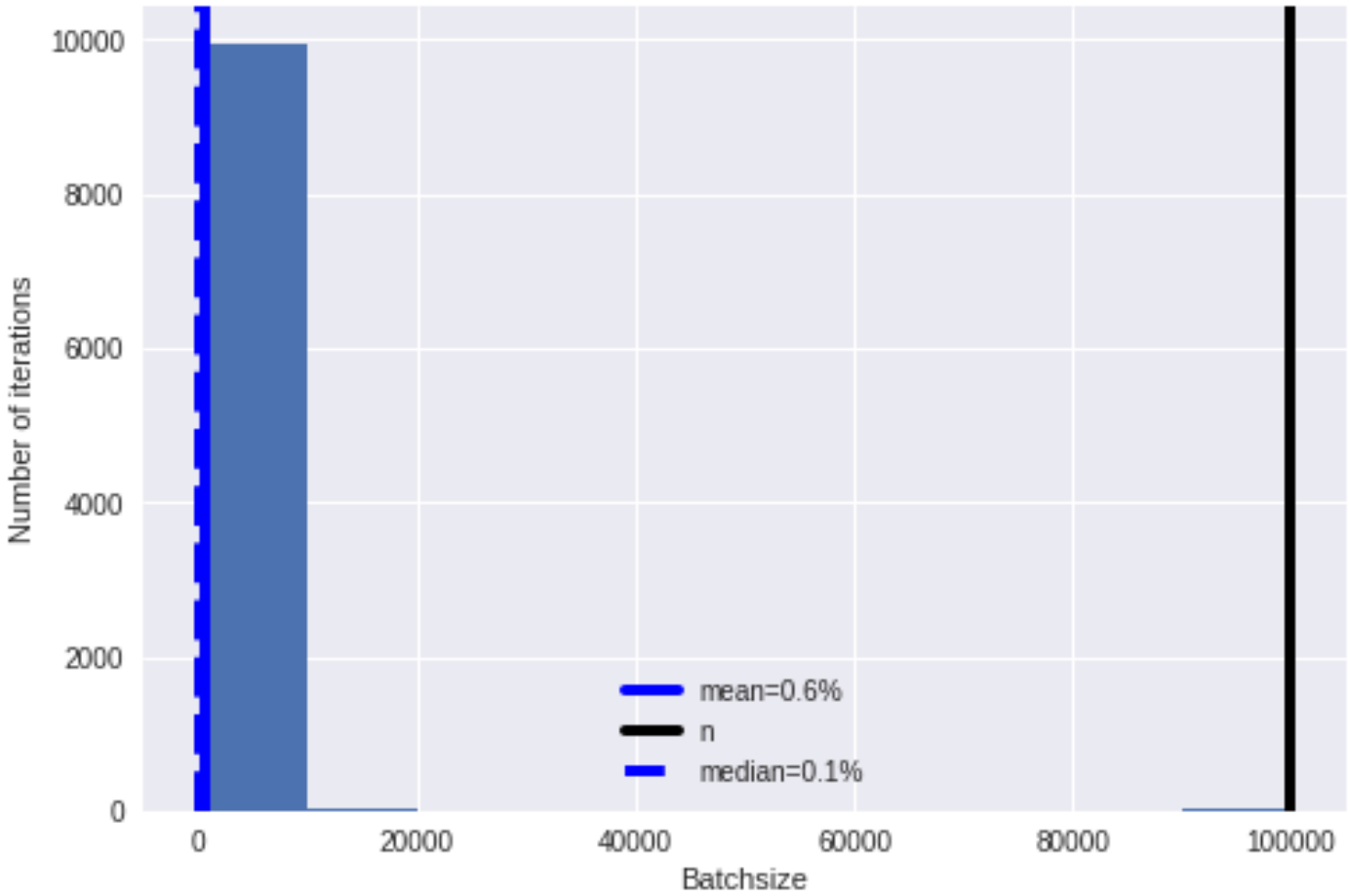
Then, the following holds :

$$\frac{1}{n} \sum_{i=1}^{n} \wp_i(\theta, \theta') = \hat{\mu}^T(\theta' - \theta) + \frac{1}{2}(\theta' - \theta)^T \hat{S}(\theta' + \theta + 2\theta_*) \approx \alpha(\theta', \theta)$$

which means that $\frac{1}{n} \sum_{i=1}^{n} \wp_i(\theta, \theta')$ can effectively be computed in $O(1)$

Example Proxy : Taylor Expansion

Modified version of the Confidence sampler with the Taylor Expansion proxy



Application : Logistic Regression

The loglikelihood is given by :

$$l_i(\theta) = \phi(y_i x_i^T \theta)$$

with : $\phi(z) = -\log(1 + e^{-z})$, y_i and x_i, the labels and the features. Then, we can use the Taylor expansion proxy with :

$$g_{i,*} = \phi'(y_i x_i^T \theta_*) y_i x_i$$

$$H_{i,*} = \phi''(y_i x_i^T \theta_*) x_i x_i^T$$

ϕ''' is bounded, we can uniformly bound $|l_i(\theta') - l_i(\theta) - \wp_i(\theta, \theta')|$:

$$|l_i(\theta') - l_i(\theta) - \wp_i(\theta, \theta')| \leq \frac{1}{24} \max_{i=1}^{n} \|x_i\|^3 (\|\theta' - \theta_*\| + \|\theta - \theta_*\|)$$

Therefore, this proxy satisfies the conditions required.

Conclusion

Sadly, unlike for gradient descent there are no "plug-and-play" approaches to scaling up MCMC methods to tall datasets :

- Divide and conquer approaches are unreliable as they have no theoretical backing for non-Gaussian settings.
- Pseudo-Marginal approximations are difficult to control and as a result are unpredictible **as is**.
- While being theoretically justified, vanilla Confidence Sampling fails to make substantial gains in computation efficiency in practice.

Conclusion

Thankfully, the scaling of MCMC methods to tall datasets is a very active area of research and novel approaches such as the use of proxies for an improved confidence sampling algorithm show promise and may provide a path forward to solving this problem.

Conclusion

Thank you !

Arnaud Doucet, Michael K 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.

Zaijing Huang and Andrew Gelman.

Sampling for bayesian computation with large datasets.

Available at SSRN 1010107, 2005.

Willie Neiswanger, Chong Wang, and Eric Xing.

Asymptotically exact, embarrassingly parallel mcmc.

arXiv preprint arXiv :1311.4780, 2013.

Steven L Scott, Alexander W Blocker, Fernando V Bonassi, Hugh A Chipman, Edward I George, and Robert E McCulloch.

Bayes and big data : The consensus monte carlo algorithm.

International Journal of Management Science and Engineering Management, 11(2) :78–88, 2016.