Gradient-Free Optimal Postprocessing of MCMC Output

by

Artem Glebov

Department of Mathematics
King's College London
The Strand, London WC2R 2LS
United Kingdom

Abstract

Contents

Introduction

Chapter 1

Background and data

1.1 Markov chain Monte Carlo

Markov chain Monte Carlo (MCMC) are a popular class of algorithms for sampling from complex probability distributions.

The need to sample from a probability distribution arises when analytical expressions are unavailable for quantities of interest, such as for example the modes or quantiles of the distribution, or for expectations with respect to the distribution, so a numerical simulation is used to obtain approximations instead. Such cases are frequent in Bayesian analysis, where the posterior density often has a complex structure with an analytically intractable normalising constant.

Describe alternatives: the inverse method, accept-reject and importance sampling

Include a simple motivating example

An MCMC algorithm proceeds by sequentially constructing a chain of samples $x_1, x_2, \ldots$, where each sample is drawn from a transition distribution Q conditional on the preceding value:

$$x_{n+1} \sim Q(x_{n+1}|x_n).$$

The distribution Q is known as the transition kernel and is selected so that it is easy to sample from and to ensure asymptotic convergence to the target distribution Π:

$$x_n \xrightarrow{d} \Pi \quad \text{as} \quad n \to \infty.$$

Two classical variations of this technique are the Metropolis-Hastings and Gibbs algorithms.

Metropolis-Hastings algorithm. The algorithm due to Metropolis et al. (1953) and Hastings (1970) uses an auxiliary proposal distribution q to sample a proposed value

$$x' \sim q(x'|x_n),$$

which is then accepted with probability

$$\alpha(x_n, x') = 1 \wedge \frac{\pi(x')}{\pi(x_n)} \frac{q(x_n|x')}{q(x'|x_n)}.$$

If x' is accepted, the algorithm sets $x_{n+1} = x'$, and otherwise $x_{n+1} = x_n$.

Consider using a different notation to avoid the confusion between the density of the proposal q and the transition kernel Q.

The common choices for the proposal distribution q are:

- A symmetric proposal satisfying $q(x'|x_n) = q(x_n|x')$, so that the ratio of the two quantities disappears from the expression for the acceptance probability:

  $$\alpha(x_n, x') = 1 \wedge \frac{\pi(x')}{\pi(x_n)}.$$

  In the special case where $q(x'|x_n) = q(x' - x_n)$ we obtain a random walk proposal:

  $$x' = x_n + Z,$$

  where Z is the distribution of the step taken by the algorithm, e.g. a multivariate normal distribution.

- An independence proposal satisfying $q(x'|x_n) = q(x')$.

Cite the ST03 lecture notes or Robert & Casella

Gibbs algorithm. Suppose x is a d-dimensional vector and the components $x^{(1)}, x^{(2)}, \ldots, x^{(d)}$ can be partitioned in such a way that we can sample the components belonging to each partition while keeping the components in

other partitions fixed. That is, let $I_i \subset \{1, \ldots, d\}$ with $\cup_{i=1}^{k} I_i = \{1, \ldots, d\}$ for some k and $I_i \cap I_j = \emptyset$ for $i \neq j$, and assume we can sample

$$x^{(I_i)} \sim f_i \left(x^{(I_i)} | x^{(I_1, \ldots, I_{i-1}, I_{i+1}, \ldots, I_k)} \right).$$

The sample x_{n+1} can then be constructed by sequentially sampling for each partition:

$$x_{n+1}^{(I_i)} \sim f_i \left(x^{(I_i)} | x_{n+1}^{(I_1, \ldots, I_{i-1})}, x_n^{(I_{i+1}, \ldots, I_k)} \right).$$

Note that the newly sampled values $x_{n+1}^{(I_1, \ldots, I_{i-1})}$ enter the computation for subsequent partitions.

Read and cite the original paper for Gibbs sampler

Consider simplifying this description

Mention HMC and other recent variations

1.2 Challenges of running MCMC

While the asymptotic convergence of MCMC samples to the target distribution is guaranteed, no general guarantee is available for finite samples however, resulting in several interrelated challenges that a practitioner faces when applying this class of algorithms:

1. The choice of a starting point for a chain affects the speed of convergence to the target distribution.

2. For a multimodal distribution, the algorithm might struggle to move between the modes within a feasible time.

3. The choice of scale of the step distribution in the random-walk Metropolis-Hastings algorithm is crucial to the algorithm's ability to explore the domain of the target distribution efficiently.

4. Assessing how close an MCMC chain is to convergence is difficult, since the knowledge about the target distribution often comes from the chain itself.

5. In order to eliminate the impact of the starting point, it can be useful to discard the initial iterations of an MCMC chain, which are considered

as "burn-in". Selecting the optimal length of the burn-in period is contingent on being able to detect convergence.

6. The sequential procedure of constructing a chain induces autocorrelation between the samples, which leads to increased variance of derived estimators.

7. The large number of samples resulting from an MCMC algorithm needs to be summarised appropriately for subsequent analysis.

The first three challenges require decisions to be made upfront before running the algorithm or adaptively during its run. In order to address the impact of the starting point, running multiple chains with starting points sampled from an overdispersed distribution is recommended (Gelman and Rubin (1992)). This approach has the added benefit of increasing the chance of discovering the modes of the target distribution, although it does not provide a guarantee in this respect.

Comparing the summary statistics of several chains offers a way to detect a lack of convergence (Gelman and Rubin (1992); Brooks and Gelman (1998); Vehtari et al. (2021)). Alternatively, the comparison can be applied to batches of samples from a single chain (Vats and Knudson (2021)). Convergence detection can be used to terminate the algorithm once a chosen criterion is satisfied, or to access the quality of the sample retrospectively.

The scaling of the step distribution in a random-walk Metropolis-Hastings algorithm is commonly tuned to target the acceptance rate of roughly $\frac{1}{4}$ for proposed samples (Gelman et al. (1996, 1997); Roberts and Rosenthal (2001)).

The last three challenges are typically addressed by post-processing a sample from a completed MCMC run. A recent proposal by **?** addresses these challenges by selecting a fixed-size subset of samples from an MCMC run such that the empirical distribution given by the subset best approximates the distribution resulting from the full sample. In the following section, we consider their approach in greater detail.

Read and cite Cowles and Carlin (1996) regarding the choice of burn-in length.

1.3 Optimal thinning as a solution to burn-in and thinning

Given a Markov chain $(X_i)_{i\in\mathbb{N}}$ and its realisation of length n, **?** set out to identify $m < n$ indices $\pi(j) \in \{1, \ldots, n\}$ such that the approximation provided by the subset of samples

$$\frac{1}{m}\sum_{j=1}^{m} \delta(X_{\pi(j)})$$

is closest to the approximation given by the full set

$$\frac{1}{n}\sum_{i=1}^{n} \delta(X_i)$$

in the sense of minimising the kernel Stein discrepancy between the two distributions.

The kernel Stein discrepancy is a special case case of an integral probability measure, which is defined for two distributions P and Q on the same measurable space $\mathcal{X}$ as

$$\mathcal{D}_{\mathcal{F}}(P, Q) \coloneqq \sup_{f\in\mathcal{F}} \left| \int_{\mathcal{X}} f \,\mathrm{d}P - \int_{\mathcal{X}} f \,\mathrm{d}Q \right|.$$

Using the Langevin Stein operator

$$\mathcal{A}_P \mathrm{g} \coloneqq p^{-1}\nabla\dot{(}p\mathrm{g})$$

for $\mathrm{g} \in \mathcal{G}$, where

$$\mathcal{G} \coloneqq \left\{ \mathrm{g} : \mathbb{R}^d \to \mathbb{R}^d \,\middle|\, \sum_{i=1}^{d} \|g_i\|^2_{\mathcal{H}(k)} \leq 1 \right\}$$

is a unit-ball in a Cartesian product of d copies of a reproducing kernel Hilbert space (RKHS) $\mathcal{H}(k)$ associated with kernel k, and taking $\mathcal{F} = \mathcal{A}_P\mathcal{G}$.

Finish up this section

1.4 Data

Describe the synthetic data

Chapter 2

Methodology

Chapter 3

Results

Chapter 4

Conclusions

Appendix A

Code

Bibliography

Stephen P. Brooks and Andrew Gelman. General Methods for Monitoring Convergence of Iterative Simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–455, December 1998.

Andrew Gelman and Donald B. Rubin. Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science*, 7(4), November 1992.

Andrew Gelman, Gareth O. Roberts, and Walter R. Gilks. Efficient Metropolis Jumping Rules. In *Bayesian Statistics 5*, pages 599–608. Oxford University PressOxford, May 1996.

Andrew Gelman, Walter R. Gilks, and Gareth O. Roberts. Weak convergence and optimal scaling of random walk Metropolis algorithms. *The Annals of Applied Probability*, 7(1), February 1997.

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

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, June 1953.

Gareth O. Roberts and Jeffrey S. Rosenthal. Optimal scaling for various Metropolis-Hastings algorithms. *Statistical Science*, 16(4), November 2001.

Dootika Vats and Christina Knudson. Revisiting the Gelman–Rubin Diagnostic. *Statistical Science*, 36(4), November 2021.

Aki Vehtari, Andrew Gelman, Daniel Simpson, Bob Carpenter, and Paul-Christian Bürkner. Rank-Normalization, Folding, and Localization: An Improved R for Assessing Convergence of MCMC (with Discussion). *Bayesian Analysis*, 16(2), June 2021.