Global consensus Monte Carlo

Lewis Rendell [1], **Adam Johansen** [1], **Anthony Lee** [2], **Nick Whiteley** [2]
L.Rendell@warwick.ac.uk, A.M.Johansen@warwick.ac.uk, Anthony.Lee@bristol.ac.uk, Nick.Whiteley@bristol.ac.uk
[1] Department of Statistics, University of Warwick, UK [2] School of Mathematics, University of Bristol, UK

Introduction

For problems involving large data sets, it may be practical or necessary to distribute the data across multiple processors. We consider a target probability density function given by

$$\pi(z) \propto \mu(z) \prod_{j=1}^{b} f_j(z)$$

where f_j is computable on processor j, requiring consideration of y_j, the jth subset of the full data set. We wish to generate samples distributed according to the corresponding distribution.

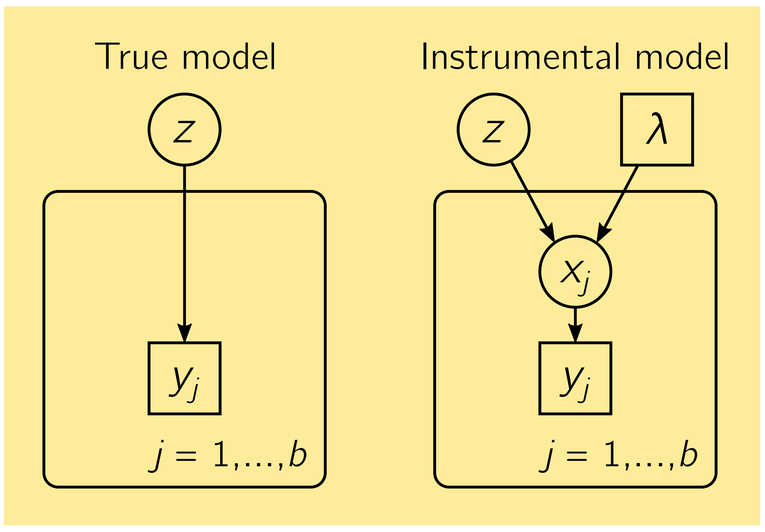
Existing approaches to this problem include:

- Scott et al. (2016), who propose running one MCMC chain on each processor, with target densities proportional to $\mu(z)^{1/b} f_j(z)$. The samples are combined in a way that implicitly assumes approximate Gaussianity.
- Xu et al. (2014), who approximate each f_j by a density belonging to an exponential family.

The instrumental model

We propose a procedure motivated by the global variable consensus optimisation algorithm of Boyd et al. (2011), itself based upon ideas of Bertsekas and Tsitsiklis (1989).

We introduce an instrumental hierarchical model by associating an instrumental variable x_j with each subset of the data, and a introducing a top-level parameter λ:



Specifically, we define a family of approximating densities on an extended state space by

$$\pi_\lambda(z, x_{1:b}) \propto \mu(z) \prod_{j=1}^{b} K_\lambda(z, x_j) f_j(x_j),$$

where $\{K_\lambda : \lambda \in \mathbb{R}_+\}$ is a family of Markov transition densities. The z-marginal of π_λ is

$$\pi_\lambda(z) \propto \mu(z) \prod_{j=1}^{b} \int K_\lambda(z, x) f_j(x) \mathrm{d}x.$$

We assume that f_j is bounded, and assume that this family satisfies $\int K_\lambda(z, x) f_j(x) \mathrm{d}x \to f_j(z)$ pointwise as $\lambda \to 0$. This implies convergence in total variation of π_λ to π, so that for bounded functions φ,

$$\int \varphi(z) \pi_\lambda(z) \mathrm{d}z \to \int \varphi(z) \pi(z) \mathrm{d}z.$$

Gibbs sampling

For given λ, a π_λ-reversible Markov chain is obtained by considering the full conditional densities:

$$\pi_\lambda(z \mid x_{1:b}) \propto \mu(z) \prod_{j=1}^{b} K_\lambda(z, x_j),$$

$$\pi_\lambda(x_j \mid z) \propto K_\lambda(z, x_j) f_j(x_j).$$

A two-variable Gibbs sampler may be constructed, where the two variables are z and $x_{1:b}$. One may thereby form estimates of $\int \varphi(z) \pi_\lambda(z) \mathrm{d}z$.

The relevance to distributed settings is that drawing a new x_j' according to the density $\pi_\lambda(x_j \mid z)$ may occur on the jth computing node; the new values $x_{1:b}'$ may then be sent to a central node that draws a new z' according to $\pi_\lambda(z \mid x_{1:b})$.

SMC sampler

The parameter λ may be chosen to balance computational tractability with fidelity to the true model, in a form of bias–variance tradeoff. To this end, a sequence $\pi_{\lambda_0}, \pi_{\lambda_1}, \ldots$ may be approximated by an SMC sampler. If π_{λ_p}-invariant MCMC kernels are used (such as those formed by the Gibbs procedure), then the potential functions depend only on the transition kernels K_λ.

By adaptively specifying the sequence of values λ_p, and using appropriate variance estimators, this approach could be used to specify λ in an automated manner.

Examples

We compare our algorithm (GCMC) with the consensus Monte Carlo algorithm (CMC) proposed by Scott et al. (2016). In both examples, we aim to estimate $\int z \pi(z) \mathrm{d}z$.

Lognormal toy example

Let $\mathcal{LN}(x; \mu, \sigma^2)$ denote the density at x of a lognormal distribution with parameters (μ, σ^2).

- Let $\mu(z) = \mathcal{LN}(z; \mu_0, \sigma_0^2)$
- Let $f_j(z) = \mathcal{LN}(z; y_j, \sigma_f^2)$
- For GCMC, use lognormal transition kernels: $K_\lambda(z, x) = \mathcal{LN}(x; \log(z), \lambda)$.

The treatment of the prior in the CMC algorithm results in large biases when an asymmetric prior is used (although a reparametrisation would solve the issue in this toy example). GCMC avoids this, but requires a careful choice of λ, as demonstrated here with $b = 10$. In each case, 10^6 samples were used, following burn-in.

	Bias	Variance
GCMC, $\lambda = 1$	1.34×10^{-1}	1.12×10^{-6}
GCMC, $\lambda = 10^{-2}$	-7.87×10^{-5}	2.88×10^{-5}
GCMC, $\lambda = 10^{-4}$	-8.55×10^{-3}	1.56×10^{-4}
CMC	$3.86 \times 10^{+3}$	2.89×10^{0}

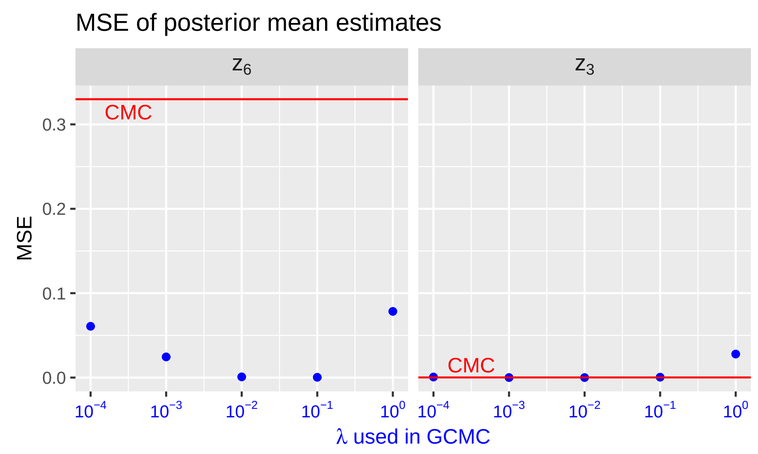
Examples (continued)

Binary regression

Binary logistic regression models are commonly used in A/B testing settings – in web design for example, to determine which content choices lead to maximised user interaction (such as the user clicking on a link to a product for sale).

- Data set formed of responses $\eta_i \in \{-1, 1\}$ and vectors $\xi_i \in \{0, 1\}^d$ of binary covariates. The data are split into b subsets; $f_j(z) = \prod_i \sigma(\eta_i z^{\mathsf{T}} \xi_i)$, where the product is taken over those indices i included in the jth data subset, and σ is the logistic function.
- We centre the covariates and use a zero-mean normal prior μ.
- For GCMC, we use normal transition kernels: $K_\lambda(z, x) \propto \mathcal{N}(x; z, \lambda)$.

We demonstrate on a simple data set with $d = 6$ covariates, split into $b = 8$ subsets, each comprising 512 data. We use various choices of λ for GCMC.



When estimating some parameters, CMC introduces a far larger bias than GCMC (if λ is chosen appropriately), as the latter is more robust to deviations from Gaussianity. In the case above left, the corresponding covariate is rarely observed in some of the data subsets; the f_j are therefore skewed in this dimension, and are poorly approximated by Gaussians.

For parameters corresponding to more frequently-observed covariates, GCMC performs comparably to CMC if λ is chosen suitably (above right).

References

Bertsekas, D. P., and Tsitsiklis, J., N. (1989). 'Parallel and distributed computation: numerical methods'. Prentice-Hall, Engelwood Cliffs.

Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J. (2011). 'Distributed optimization and statistical learning via the alternating direction method of multipliers'. Foundations and Trends in Machine Learning, 3(1):1–122.

Scott, S. L., Blocker, A. W., Bonassi, F. V., Chipman, H. A., George, E. I., and McCulloch, R. E. (2016). 'Bayes and big data: the consensus Monte Carlo algorithm'. International Journal of Management Science and Engineering Management, 11(2):78–88.

Xu, M., Teh, Y. W., Zhu, J., and Zhang, B. (2014). 'Distributed context-aware Bayesian posterior sampling via expectation propagation'. In: Advances in Neural Information Processing Systems, pages 3356–3364.

Acknowledgements: Lewis Rendell is funded by EPSRC grant number EP/M508184/1.