Computational Statistics and Data Analysis 83 (2015) 101–115

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Adaptive Metropolis algorithm using variational Bayesian adaptive Kalman filter

Isambi S. Mbalawata [a,*], Simo Särkkä [b], Matti Vihola [c], Heikki Haario [a]

[a] Department of Mathematics and Physics, Lappeenranta University of Technology, P.O. Box 20, FI-53851 Lappeenranta, Finland
[b] Department of Biomedical Engineering and Computational Science, Aalto University, P.O. Box 12200, FI-00076 Aalto, Finland
[c] Department of Statistics, University of Oxford, 1 South Parks Road, Oxford, OX1 3TG, United Kingdom

ARTICLE INFO

Article history:
Received 27 August 2013
Received in revised form 1 October 2014
Accepted 5 October 2014
Available online 16 October 2014

Keywords:
Markov chain Monte Carlo
Adaptive Metropolis algorithm
Adaptive Kalman filter
Variational Bayes

ABSTRACT

Markov chain Monte Carlo (MCMC) methods are powerful computational tools for analysis of complex statistical problems. However, their computational efficiency is highly dependent on the chosen proposal distribution, which is generally difficult to find. One way to solve this problem is to use adaptive MCMC algorithms which automatically tune the statistics of a proposal distribution during the MCMC run. A new adaptive MCMC algorithm, called the variational Bayesian adaptive Metropolis (VBAM) algorithm, is developed. The VBAM algorithm updates the proposal covariance matrix using the variational Bayesian adaptive Kalman filter (VB-AKF). A strong law of large numbers for the VBAM algorithm is proven. The empirical convergence results for three simulated examples and for two real data examples are also provided.

© 2014 Elsevier B.V. All rights reserved.

1. Introduction

Markov chain Monte Carlo (MCMC) methods (Brooks et al., 2011) are an important class of numerical tools for approximating multidimensional integrals over complicated probability distributions in Bayesian computations and in various other fields. The computational efficiency of MCMC sampling depends on the choice of the proposal distribution. A challenge of MCMC methods is that in complicated high-dimensional models it is very hard to find a good proposal distribution.

The Gaussian distribution is often used as a proposal distribution due to its theoretical and computational properties. However, the Gaussian proposal distribution needs a well tuned covariance matrix for optimal acceptance rate and good mixing of the Markov chain. If the covariance matrix is too small, too large or has improper correlation structure, the Markov chains will be highly positively correlated and hence the estimators will have a large variance. Because manual tuning is laborious, several adaptive MCMC algorithms have been suggested (Haario et al., 1999, 2001, 2006; Vihola, 2012; Andrieu and Thoms, 2008; Roberts and Rosenthal, 2007, 2009; Atchadé and Rosenthal, 2005; Gelman et al., 1996) to update the covariance matrix during the MCMC run.

In this article, we propose a new adaptive Metropolis algorithm, where we update the covariance matrix of the Gaussian proposal distribution of the Metropolis algorithm using the variational Bayesian adaptive Kalman filter (VB-AKF) proposed by Särkkä and Nummenmaa (2009) and Särkkä and Hartikainen (2013). The idea of the classical Metropolis algorithm (Haario et al., 1999) is essentially to empirically estimate the covariance of the samples and use this estimate to construct the proposal distribution. However, as we point out here, such a covariance estimation problem can also be formulated

* Corresponding author.
E-mail address: Isambi.Mbalawata@lut.fi (I.S. Mbalawata).

http://dx.doi.org/10.1016/j.csda.2014.10.006
0167-9473/© 2014 Elsevier B.V. All rights reserved.

as an instance of recursive Bayesian estimation, where the term used for this kind of recursive estimation is *Bayesian filtering* (Särkkä, 2013). This reinterpretation allows one to construct alternative and potentially more effective adaptation mechanisms by utilizing the various Bayesian filtering algorithms developed over the years for doing the covariance estimation. The aim of this article is to propose a practical algorithm which is constructed from this underlying idea, prove its convergence, and test its performance empirically.

The structure of this article is the following: in Section 2 we review the existing adaptive MCMC methods. Section 3 is dedicated to our new adaptive Metropolis algorithm. Theoretical validity of the proposed algorithm is shown in Section 4 by proving a strong law of large numbers. In Section 5, we study the empirical convergence of the method in three simulated examples and then apply the method to two real data examples.

2. Adaptive Markov chain Monte Carlo methods

Markov chain Monte Carlo (MCMC) methods are widely used algorithms for drawing samples from complicated multidimensional probability distributions. For example, in Bayesian analysis (Gelman et al., 2013), we are often interested in computing the posterior expectation of a function $\mathbf{g}(\boldsymbol{\theta})$ given the measurements $\mathbf{z}_1, \ldots, \mathbf{z}_M$:

$$\mathrm{E}[\mathbf{g}(\theta) \mid \mathbf{z}_1, \ldots, \mathbf{z}_M] = \int_{\mathbb{R}^d} \mathbf{g}(\theta)\, p(\theta \mid \mathbf{z}_1, \ldots, \mathbf{z}_M)\, \mathrm{d}\theta. \tag{1}$$

We can use MCMC methods to approximate the expectation by drawing samples from the posterior distribution

$$\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots, \boldsymbol{\theta}_n \sim p(\boldsymbol{\theta} \mid \mathbf{z}_1, \ldots, \mathbf{z}_M), \tag{2}$$

and then by employing the approximation

$$\mathrm{E}[\mathbf{g}(\boldsymbol{\theta}) \mid \mathbf{z}_1, \ldots, \mathbf{z}_M] \approx \frac{1}{n} \sum_{i=1}^{n} \mathbf{g}(\boldsymbol{\theta}_i). \tag{3}$$

A common construction for MCMC uses a random walk that explores the state space through local moves. The most well-known traditional MCMC method is the *Metropolis algorithm*. In the Metropolis algorithm we draw a candidate point $\boldsymbol{\theta}_*$ from a symmetric proposal distribution $q(\boldsymbol{\theta}_* \mid \boldsymbol{\theta})$ and use an accept/reject rule to accept or reject the sampled point (Gilks et al., 1996; Gelman et al., 2013; Brooks et al., 2011).

The efficiency of an MCMC algorithm can be improved by carefully tuning the proposal distribution. Adaptive MCMC methods are a family of algorithms, which take care of the tuning automatically. This proposal is often chosen to be a Gaussian distribution, in which case it is the covariance matrix that needs to be tuned. Under certain settings Gelman et al. (1996) show that the optimal covariance matrix for an MCMC algorithm with Gaussian proposal is $\lambda\, \boldsymbol{\Sigma}$, with $\lambda = 2.38^2/d$, where d is the dimension and $\boldsymbol{\Sigma}$ is the $d \times d$ covariance matrix of the target distribution.

In the *adaptive Metropolis (AM) algorithm* by Haario et al. (2001), the covariance matrix $\boldsymbol{\Sigma}_{k-1}$ for the step k is estimated as follows:

$$\boldsymbol{\Sigma}_{k-1} = \mathrm{cov}(\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{k-1}) + \varepsilon \mathbf{I}, \tag{4}$$

where $\mathbf{I}$ is the $d \times d$ identity matrix and ε is a small positive value whose role is to make sure that $\boldsymbol{\Sigma}_{k-1}$ is not singular (Haario et al., 1999, 2001). The AM algorithm of Haario et al. (2001) can be summarized as follows:

- Initialize $\boldsymbol{\theta}_0$, $\boldsymbol{\Sigma}_0$.
- For $k = 1, 2, 3, \ldots$
  - Sample a candidate point $\boldsymbol{\theta}_*$ from a Gaussian distribution
    $$\boldsymbol{\theta}_* \sim \mathbb{N}(\boldsymbol{\theta}_{k-1}, \lambda\, \boldsymbol{\Sigma}_{k-1}). \tag{5}$$
  - Compute the acceptance probability
    $$\alpha_k = \min\left\{1,\ \frac{p(\boldsymbol{\theta}_* \mid \mathbf{z}_1, \ldots, \mathbf{z}_M)}{p(\boldsymbol{\theta}_{k-1} \mid \mathbf{z}_1, \ldots, \mathbf{z}_M)}\right\}. \tag{6}$$
  - Sample a random variable u from the uniform distribution $\mathbb{U}(0, 1)$.
  - If $u < \alpha_k$, set $\boldsymbol{\theta}_k = \boldsymbol{\theta}_*$. Otherwise set $\boldsymbol{\theta}_k = \boldsymbol{\theta}_{k-1}$.
  - Compute the covariance matrix $\boldsymbol{\Sigma}_k$ using Eq. (4).

Different adaptive algorithms have been proposed as improved versions of the AM algorithm above. Good surveys of such algorithms are found in Andrieu and Thoms (2008) and Liang et al. (2010), where the authors present ways to implement the algorithms and then show why the algorithms preserve the correct stationary distributions. For instance, apart from updating the covariance alone, one can adapt λ using the following Robbins–Monro algorithm, which alleviates the problem of $\boldsymbol{\Sigma}_k$ being systematically too large or too small (Andrieu and Thoms, 2008; Atchadé and Fort, 2010; Vihola, 2011):

$$\log(\lambda_k) = \log(\lambda_{k-1}) + \gamma_k\, (\alpha_k - \overline{\alpha}). \tag{7}$$

In Eq. (7), $\overline{\alpha}$ is the target acceptance rate which is commonly set to 0.234 and γ_k is a gain factor sequence satisfying the following conditions:

$$\sum_{k=1}^{\infty} \gamma_k = \infty \quad \text{and} \quad \sum_{k=1}^{\infty} \gamma_k^{1+\delta} < \infty \quad \text{for some } \delta \in (0, 1].$$

Download English Version:

https://daneshyari.com/en/article/414940

Download Persian Version:

https://daneshyari.com/article/414940

Daneshyari.com