2011 International Symposium on Nonlinear Theory and its Applications
NOLTA2011, Kobe, Japan, September 4-7, 2011

An Interacting Particle Method for Approximate Bayes Computations

C. Albert

Eawag, aquatic research, 8600 Dübendorf, Switzerland; carlo.albert@eawag.ch

Abstract— Approximate Bayes computations are used for parameter inference when the likelihood function is expensive to calculate but relatively cheap to sample from. We present a new interacting particle method for approximate Bayes computations. Unlike other algorithms, it is not based on importance sampling. Hence, it does not suffer from a loss of effective sample size due to re-sampling.

1. Introduction

One way of doing parameter inference in the Bayesian framework is to generate samples from the *posterior distribution*

$$f_{post}(\boldsymbol{\theta}|\mathbf{y}) = \frac{f(\mathbf{y}|\boldsymbol{\theta})f(\boldsymbol{\theta})}{f(\mathbf{y})}\,, \tag{1}$$

where $f(\boldsymbol{\theta})$ denotes the *prior distribution* encoding our knowledge about the parameter vector $\boldsymbol{\theta}$ before the experiment and $f(\mathbf{y}|\boldsymbol{\theta})$ is the *likelihood function*, that is, the probability density of outputs given the parameter vector $\boldsymbol{\theta}$, evaluated at the measurement vector $\mathbf{y}$. Numerical methods such as *Metropolis-Hastings* require many evaluations of the likelihood function to generate such a sample. However, for truly stochastic models, the likelihood function is often prohibitively expensive to calculate. Therefore, in recent years, algorithms have been suggested that generate samples from (1) and are based on *sampling from* the likelihood rather than calculating it.

As far as we know, the origin of these algorithms is to be found in population genetics. Pritchard et al. [5] used an algorithm that generates samples from the joint distribution $f(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})$ and accepts the simulated $\boldsymbol{\theta}$ only if $\mathbf{x} = \mathbf{y}$. For continuous outputs (or output spaces of high cardinality), this equality condition needs to be relaxed. Therefore, a metric, ρ, on the output space was introduced and algorithms were invented that sample from the probability distribution proportional to $f(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})\chi(\rho(\mathbf{x},\mathbf{y}) < \epsilon)$, that is, an approximate posterior. This is why these algorithms are often called *Approximate Bayes Computations* (ABC). Marjoram et al. [4] used *Markov chains* to produce samples from an approximate posterior. Their algorithm combines a random walk in parameter space with drawing from the likelihood and an acceptance/rejection step that accounts for the prior and only accepts moves into an ϵ band around the target $\mathbf{y}$. However, a small static tolerance leads to a high rejection rate. Therefore, Beaumont et al. [1] allowed for a decreasing sequence of tolerances and let a population of particles of constant size N evolve towards a good approximation of the posterior. Their algorithm consists of an iteration of *importance sampling* steps, that is, each iteration consists of drawing a new population from the old one with weights and subsequent re-weighting. This re-weighting leads to a loss of effective sample size at each step and, furthermore, computational costs of the order $\mathcal{O}(N^2)$. Their algorithm also uses the empirical variances of the population to adapt the jump distribution in parameter space.

In this short paper, we present a new population method that is of the order $\mathcal{O}(N)$ and does not suffer from a loss of effective sample size. The idea is to run N parallel Markov chains that combine a random walk in parameter space with drawing from the likelihood and an acceptance/rejection step. Moves are more likely to be accepted if they go into a region of higher prior density and if they move closer to the target $\mathbf{y}$. This way, each chain produces samples from the approximate posterior $f(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})\exp(-\rho(\mathbf{x},\mathbf{y})/\epsilon)$. Both the jump distribution in parameter space and the tolerance ϵ are adapted using the empirical covariance of the population in parameter space and its average distance from the target, respectively. The adaptations of the tolerance ϵ and the jump distribution in parameter space render the underlying stochastic process non-linear. However, these adaptations can be interpreted as *mean-field interactions* between the particles of the population. Due to this fact, stability and uniqueness of the limiting distribution can be proven with a slight adaptation of the proof of the H-theorem from statistical physics. Furthermore, the particles remain statistically *independent*, in the limit of an infinite sample size.

For high dimensions, n, of the output space, the tolerance ϵ that can be achieved in reasonable time is limited. This deficiency is inherent to all ABC algorithms simply because drawing an output from an ϵ-ball around $\mathbf{y}$ scales like ϵ^n. Methods to reduce this bias are investigated elsewhere (see, e.g., Leuenberger

et al. [3]).

A convergence proof for the algorithm, case studies as well as an estimate for the scaling behavior of the bias with the dimension of the output space will be presented elsewhere.

2. Algorithm

Our aim is to sample, in an efficient manner, from the posterior distribution (1). Instead of sampling directly from the posterior, like in the Metropolis-Hastings algorithm, we may rewrite it as the marginalization

$$f_{post}(\boldsymbol{\theta}|\mathbf{y}) \propto \int f(\mathbf{x}|\boldsymbol{\theta}) f(\boldsymbol{\theta}) \delta(\mathbf{x}-\mathbf{y}) d\mathbf{x} \qquad (2)$$

and sample from the joint distribution $f(\mathbf{x}|\boldsymbol{\theta})f(\boldsymbol{\theta})\delta(\mathbf{x}-\mathbf{y})$ in the $(\boldsymbol{\theta},\mathbf{x})$-space, $\Theta \times Y$. If the output space has a high cardinality or is continuous, sampling from $f(\mathbf{x}|\boldsymbol{\theta})\delta(\mathbf{x}-\mathbf{y})$ becomes inefficient or impossible, respectively. In these cases, we approximate the delta function by a sequence of distributions on the output space Y that are ever sharper peaked around the measured output $\mathbf{y}$ until a satisfactory level of precision is reached. To this end, we choose a *metric*

$$\rho(\cdot,\cdot) \,:\, Y \times Y \longrightarrow \mathbb{R}_0^+ \qquad (3)$$

with the properties that

$$\rho(\mathbf{x},\mathbf{y}) = 0 \quad \text{iff} \quad \mathbf{x} = \mathbf{y}\,,$$

and

$$\int \exp(-\rho(\mathbf{x},\mathbf{y}))d\mathbf{x} < \infty\,.$$

We then have that

$$\lim_{\epsilon \searrow 0} \int f(\boldsymbol{\theta}) f(\mathbf{x}|\boldsymbol{\theta}) e^{-\rho(\mathbf{x},\mathbf{y})/\epsilon} d\mathbf{x} \propto f(\boldsymbol{\theta}) f(\mathbf{y}|\boldsymbol{\theta})\,.$$

The aim is to propagate a population of particles in $\Theta \times Y$ that represents a sample from a time-dependent distribution, $W(\boldsymbol{\theta},\mathbf{x};t)$, which converges towards $f(\boldsymbol{\theta})f(\mathbf{x}|\boldsymbol{\theta})\exp(-\rho(\mathbf{x},\mathbf{y})/\epsilon)$, for a sufficiently small tolerance ϵ. Therefore, each particle is propagated according to the *transition rate*

$$\begin{aligned} t(\boldsymbol{\theta},\mathbf{x}|\boldsymbol{\theta}',\mathbf{x}') &= k(\boldsymbol{\theta}|\boldsymbol{\theta}') f(\mathbf{x}|\boldsymbol{\theta}) \\ &\quad \times \min\left(1, \exp\left(\frac{\rho(\mathbf{x}',\mathbf{y}) - \rho(\mathbf{x},\mathbf{y})}{\epsilon}\right)\right) \\ &\quad \times \min\left(1, \frac{f(\boldsymbol{\theta})}{f(\boldsymbol{\theta}')}\right)\,, \end{aligned} \qquad (4)$$

where $k(\boldsymbol{\theta}|\boldsymbol{\theta}')$ is a symmetric transition rate in Θ. Algorithmically, (4) is implemented composing a random walk in parameter space with drawing from the likelihood and an *acceptance/rejection* step to account for prior and ϵ-tolerance.

The time-course of $W(\boldsymbol{\theta},\mathbf{x},t)$ is described by the *master equation*

$$\begin{aligned} \frac{\partial}{\partial t} W(\boldsymbol{\theta},\mathbf{x},t) = \int &(t(\boldsymbol{\theta},\mathbf{x}|\boldsymbol{\theta}',\mathbf{x}')W(\boldsymbol{\theta}',\mathbf{x}',t) \\ &- t(\boldsymbol{\theta}',\mathbf{x}'|\boldsymbol{\theta},\mathbf{x})W(\boldsymbol{\theta},\mathbf{x},t))d\boldsymbol{\theta}'d\mathbf{x}'\,. \end{aligned} \qquad (5)$$

It is easy to verify that

$$W_{eq}(\boldsymbol{\theta},\mathbf{x}) \propto f(\boldsymbol{\theta}) f(\mathbf{x}|\boldsymbol{\theta}) \exp\left(-\frac{\rho(\mathbf{x},\mathbf{y})}{\epsilon}\right) \qquad (6)$$

represents a *stationary solution* of (5). Moreover, this solution exhibits *detailed balance*

$$t(\boldsymbol{\theta},\mathbf{x}|\boldsymbol{\theta}',\mathbf{x}')W_{eq}(\boldsymbol{\theta}',\mathbf{x}') = t(\boldsymbol{\theta}',\mathbf{x}'|\boldsymbol{\theta},\mathbf{x})W_{eq}(\boldsymbol{\theta},\mathbf{x})\,, \qquad (7)$$

i.e., the integrand on the r.h.s. of (5) vanishes at stationarity. This is why W_{eq} is also called an *equilibrium distribution* (see [6], 6.4, for the terminology).

In order to improve the efficiency of the algorithm, ϵ is replaced by a decreasing function $\epsilon(t)$ that is adapted to the relaxation velocity of the population towards equilibrium. If $\epsilon(t)$ decreaes much slower than the relaxation velocity the algorithm is clearly inefficient; if it decreases much faster, too many particles "get stuck" and the algorithm becomes inefficient as well. At equilibrium, for the standard metric on $\mathbb{R}^n$ and an ϵ that is so small that the variation of $f(\mathbf{x}|\boldsymbol{\theta})$ within an ϵ-ball around $\mathbf{y}$ can be neglected, $\rho(\mathbf{x},\mathbf{y})$ is approximately Γ-distributed with shape parameter n and scale parameter ϵ. Hence, we have that

$$\langle \rho(\mathbf{x},\mathbf{y}) \rangle_{W_{eq}} \approx n\epsilon\,.$$

Therefore, we use the average of $\rho(\mathbf{x},\mathbf{y})$ over the population to adapt ϵ and replace ϵ in (4) by

$$\epsilon_W = \frac{1}{\beta_1 n} \langle \rho(\mathbf{x},\mathbf{y}) \rangle_W\,, \qquad (8)$$

where β_1 may be interpreted as an inverse temperature w.r.t. the output space.

We also adapt the jump distribution $k(\boldsymbol{\theta}|\boldsymbol{\theta}')$ to the empirical covariance of the population. Therefore, we replace $k(\boldsymbol{\theta}|\boldsymbol{\theta}')$ in (4) by

$$k_W(\boldsymbol{\theta}|\boldsymbol{\theta}') \propto \exp\left(-\frac{\beta_2}{2}(\boldsymbol{\theta}-\boldsymbol{\theta}')^T(\Sigma_W + s\mathbb{1})^{-1}(\boldsymbol{\theta}-\boldsymbol{\theta}')\right)\,, \qquad (9)$$

where Σ_W denotes the covariance matrix of W and s is a small scalar that prevents $k_W(\boldsymbol{\theta}|\boldsymbol{\theta}')$ from degenerating. Furthermore, β_2 may be interpreted as an inverse temperature w.r.t. the parameter space.

Despite the non-linearities (8) and (9), the standard H-theorem of master equations can be applied to prove convergence (see, e.g., [2], for a proof of the H-theorem in the linear case), as long as β_1 is sufficiently large for ϵ_W to be monotonously decreasing.

For convenience, we conclude with a description of the algorithm in pseudo-code:

1. Initialization of the algorithm:

   (a) Draw initial population of particles, $\boldsymbol{\theta}_{0,i}$, for $i = 1, \ldots, N$, from the prior.

   (b) Draw an output, $\mathbf{x}_{0,i}$, for each particle, from the likelihood $f(\mathbf{x}|\boldsymbol{\theta}_{0,i})$.

   (c) Calculate $\rho_{0,i} = \rho(\mathbf{x}_{0,i}, \mathbf{y})$, for $i = 1, \ldots, N$.

   (d) Set iteration counter $k = 0$.

2. Iterate the following steps:

   (a) Increment k.

   (b) Adapt tolerance:

$$\epsilon_k = \frac{1}{\beta_1 N n} \sum_{i=1}^{N} \rho_{k-1,i} \, .$$

   (c) Adapt jump covariance:

$$\Sigma_{k,ab} = \frac{1}{\beta_2 (N-1)} \sum_{i=1}^{N} (\theta^a_{k-1,i} - \bar{\theta}^a_{k-1,i})(\theta^b_{k-1,i} - \bar{\theta}^b_{k-1,i}) + s\delta_{ab} \, ,$$

   where

$$\bar{\theta}^a_{k-1,i} = \frac{1}{N} \sum_{i=1}^{N} \theta^a_{k-1,i} \, .$$

   For $i = 1, \ldots, N$ do the following steps

   i. Draw proposal parameter vectors from normal distributions

$$\boldsymbol{\theta}^*_{k,i} \sim N(\boldsymbol{\theta}_{k-1,i}, \Sigma_k) \, .$$

   ii. Draw outputs, $\mathbf{x}^*_{k,i}$, from the likelihood $f(\mathbf{x}|\boldsymbol{\theta}^*_{k,i})$.

   iii. Calculate $\rho^*_{k,i} = \rho(\mathbf{x}^*_{k,i}, \mathbf{y})$.

   iv. Draw random number, r, from uniform distribution $\mathcal{U}[0, 1]$.

   v. If

$$r < \min\left(1, \exp\left(\frac{\rho_{k-1,i} - \rho^*_{k,i}}{\epsilon_k}\right)\right) \times \min\left(1, \frac{f(\boldsymbol{\theta}^*_{k,i})}{f(\boldsymbol{\theta}_{k-1,i})}\right)$$

   set $\boldsymbol{\theta}_{k,i} = \boldsymbol{\theta}^*_{k,i}$, $\mathbf{x}_{k,i} = \mathbf{x}^*_{k,i}$ and $\rho_{k,i} = \rho^*_{k,i}$. Otherwise, set $\boldsymbol{\theta}_{k,i} = \boldsymbol{\theta}_{k-1,i}$, $\mathbf{x}_{k,i} = \mathbf{x}_{k-1,i}$ and $\rho_{k,i} = \rho_{k-1,i}$.

References

[1] Beaumont M.A., *Adaptive approximate Bayesian computation*, Biometrika **96**, 4, 983-990, (2009).

[2] N.G. van Kampen, *Stochastic processes in physics and chemistry*, 3rd ed. Amsterdam: Elsevier 2007.

[3] Leuenberger C., Wegmann D., *Bayesian Computation and Model Selection Without Likelihoods*, Genetics **184**, 243-252 (2010).

[4] Marjoram P., Molitor J., Plagnol V., Tavaré S., *Markov chain Monte Carlo without likelihoods*, Proc. Natl. Acad. Sci. U.S.A. **100**, 15324-15328 (2003).

[5] Pritchard J. K., Seielstad M. T., Perez-Lezaun A., Fledman M. W., *Population growth of human Y chromosomes: a study of Y chromosome microsatellites*, Molec. Biol. Evol. **16**, 1791-1798 (1999).

[6] Risken, H. *The Fokker-Planck Equation*, 2nd Ed., Springer Series in Synergetics, Springer-Verlag Berlin Heidelberg New York (1989).