ABC-EP: Expectation Propagation for Likelihood-free Bayesian Computation

Simon Barthelmé SIMON.BARTHELME@BCCN-BERLIN.DE
Modelling of Cognitive Processes, Bernstein Center for Computational Neuroscience and TU Berlin. Franklinstr. 28/29, 10961 Berlin, Germany.

Nicolas Chopin NICOLAS.CHOPIN@ENSAE.FR
ENSAE-CREST, 3, Avenue Pierre Larousse, 92240 Malakoff, France

Abstract

Many statistical models of interest to the natural and social sciences have no tractable likelihood function. Until recently, Bayesian inference for such models was thought infeasible. Pritchard et al. (1999) introduced an algorithm known as *ABC,* for *Approximate Bayesian Computation*, that enables Bayesian computation in such models. Despite steady progress since this first breakthrough, such as the adaptation of MCMC and Sequential Monte Carlo techniques to likelihood-free inference, state-of-the art methods remain hard to use and require enormous computation times. Among other issues, one faces the difficult task of finding appropriate summary statistics for the model, and tuning the algorithm can be time-consuming when little prior information is available. We show that Expectation Propagation, a widely successful approximate inference technique, can be adapted to the likelihood-free context. The resulting algorithm does not require summary statistics, is an order of magnitude faster than existing techniques, and remains usable when prior information is vague.

1. Introduction

Nearly all methods in Bayesian analysis and statistical machine learning presuppose that the likelihood function can be efficiently calculated. However, for a wide range of models this is simply not the case: although given a set of parameters $\boldsymbol{\theta}$ the model can be simulated to yield some output $\mathbf{y}$, one cannot easily compute the probability $p(\mathbf{y}|\boldsymbol{\theta})$ of observing a particular output for a particular set of parameters, most often because this would involve an intractable integral over latent variables. Examples can be found across a wide range of fields, from biology (Beaumont, 2010) to economics (Gouriéroux et al., 1993), and the underlying model does not need to be complex - indeed, some of the simplest queuing models are intractable in that way (Blum & François, 2010).

We may still want to perform Bayesian inference on such a model: this involves being able to compute some expectations over the posterior, and, often, being able to compute the evidence (marginal likelihood) for the model. The rejection ABC algorithm of Pritchard et al. (1999) provides a way to do so by producing samples from an approximation to the posterior distribution. Rejection ABC is simple to state and understand and forms the basis for all likelihood-free inference methods.

We have data $\mathbf{y}^\star$, a model for the data $p_l(\mathbf{y}^\star|\boldsymbol{\theta})$, a prior over the parameters $p_0(\boldsymbol{\theta})$, and we would like to obtain samples from the posterior distribution:

$$p(\boldsymbol{\theta}|\mathbf{y}^\star) \propto p_0(\boldsymbol{\theta})p_l(\mathbf{y}^\star|\boldsymbol{\theta})$$

If we assume that the likelihood can be sampled from, then the following procedure produces exact samples from the posterior.

1. Draw $\boldsymbol{\theta}$ from the prior.
2. Draw a dataset $\mathbf{y}$ from the model conditional on $\boldsymbol{\theta}$, $\mathbf{y} \sim p(\mathbf{y}|\boldsymbol{\theta})$.
3. If $\mathbf{y} = \mathbf{y}^\star$ then keep $\boldsymbol{\theta}$, otherwise reject.

Appearing in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

Evidently as a practical algorithm the scheme has two major flaws: it only works for discrete $\mathbf{y}$, and as the dimension of $\mathbf{y}$ increases the acceptance rate drops exponentially. Rejection ABC remedies these problems by introducing a vector of summary statistics $\mathbf{s}(\mathbf{y})$, and accepting samples if the summary statistics of the simulated dataset are within a distance ϵ from the summary statistics of the true dataset. This produces samples from an approximation of the posterior given by

$$p_{\mathbf{s},\epsilon}(\boldsymbol{\theta}|\mathbf{y}^\star) \propto \int_{\mathcal{Y}} p_0(\boldsymbol{\theta})\, p_l(\mathbf{y}|\boldsymbol{\theta}) k_\epsilon(||\mathbf{s}(\mathbf{y}) - \mathbf{s}(\mathbf{y}^\star)||)\, \mathrm{d}\mathbf{y} \tag{1}$$

where k_ϵ is a window of width ϵ, defining the acceptance region. It is important to stress that, unless $\mathbf{s}$ is sufficient (as in example 1 below), the approximation error does not vanish as $\epsilon \to 0$, i.e. $p(\boldsymbol{\theta}|\mathbf{s}(\mathbf{y}^\star)) \neq p(\boldsymbol{\theta}|\mathbf{y}^\star)$. In that respect, the ABC posterior suffers from two levels of approximation: a non-parametric error governed by the bandwidth ϵ, and a bias introduced by the summary statistics $\boldsymbol{s}$. The more we include in $\mathbf{s}(\mathbf{y})$, the smaller the bias induced by the dimensionality reduction should be. On the other hand, as the dimensionality of $\mathbf{s}(\mathbf{y})$ increases, the lower the acceptance rate will be. We would then have to increase ϵ, which leads to an approximation of lower quality.

We describe below an adaptation of Expectation Propagation (Minka, 2001), which we call the ABC-EP algorithm. ABC-EP simplifies considerably likelihood free inference in three ways: it does not require summary statistics, enables inference even when prior constraints are vague, and reduces computation time dramatically.

2. Related work

Although the simplicity of the ABC-rejection algorithm is appealing, the very low acceptance rate one encounters in practice limits its use. Marjoram et al. (2003) therefore suggested adapting the Metropolis-Hastings algorithm, yielding a method we will call MCMC-ABC. MCMC-ABC targets the joint distribution $p(\boldsymbol{\theta}, \mathbf{y})$ truncated such that $||\mathbf{s}(\mathbf{y}) - \mathbf{s}(\mathbf{y}^\star)|| < \epsilon$. This is done by generating candidate $(\boldsymbol{\theta}_{t+1}, \mathbf{y}_{t+1})$ where $\mathbf{y}_{t+1} \sim p_l(\mathbf{y}|\boldsymbol{\theta}_t)$, and accepting only if $||\mathbf{s}(\mathbf{y}_{t+1}) - \mathbf{s}(\mathbf{y}^\star)|| < \epsilon$. This algorithm requires the simulation of a full dataset at each iteration.

Del Moral et al. (2008) adapt sequential Monte Carlo techniques (Del Moral et al., 2006) to likelihood-free problems (ABC-SMC). In ABC-SMC a population of particles is updated over time to approximate a sequence of ABC posteriors $p_{\mathbf{s},\epsilon_1}(), \ldots, p_{\mathbf{s},\epsilon_k}()$, with $\epsilon_1 > \ldots > \epsilon_k$. Since for ϵ large the ABC posterior tends to the prior, the intuition is that decreasing ϵ gradually lets one approach the target density relatively smoothly.

Another line of work views likelihood-free inference as a density estimation problem, in which we can sample from the joint density $p(\boldsymbol{\theta}, \mathbf{y})$ and we want to estimate the density of $\boldsymbol{\theta}$ conditional on $\mathbf{y}^\star$ (Blum, 2010). Several "correction" algorithms have been proposed, including Beaumont et al. (2002) and Blum & François (2010). Finally, other methods focus on selecting appropriate summary statistics (Nunes & Balding, 2010). A fairly extensive review of existing techniques can be found in Beaumont (2010).

3. The ABC-EP algorithm

3.1. Expectation Propagation

Expectation Propagation (introduced in Minka, 2001 and closely related to the Expectation Consistent inference of Opper & Winther, 2005) is one of the most successful algorithms for variational inference. It aims at finding a tractable density $q(\boldsymbol{\theta}) \in \mathcal{Q}$ that is close to a target density $p(\boldsymbol{\theta})$ in the sense of minimising the Kullback-Leibler $KL(p||q) = \int p(\boldsymbol{\theta}) \log \frac{p(\boldsymbol{\theta})}{q(\boldsymbol{\theta})} \mathrm{d}\boldsymbol{\theta}$. - Here we use multivariate Gaussian distributions as approximating distributions, but other families can be used. Most generally one takes the approximating family $\mathcal{Q}$ to be an exponential family parameterised by :

$$q_{\boldsymbol{\lambda}}(\boldsymbol{\theta}) = \exp\left(\boldsymbol{\lambda}^{\mathbf{t}} \boldsymbol{t}(\boldsymbol{\theta}) - \phi(\boldsymbol{\lambda})\right)$$

Then $\boldsymbol{\lambda}_{opt} = \underset{\boldsymbol{\lambda}}{\operatorname{argmin}} KL(p||q_{\boldsymbol{\lambda}})$ is such that $\mathbb{E}(\mathbf{t}(\boldsymbol{\theta}))$ under $q_{\boldsymbol{\lambda}_{opt}}$ is equal to $\mathbb{E}(\mathbf{t}(\boldsymbol{\theta}))$ under p (Seeger, 2005) - the objective of EP can be then understood equivalently as computing a set of moments of p. If q is a multivariate Gaussian, then $q_{\boldsymbol{\lambda}_{opt}}$ will have the same mean and covariance as p.

Although $\boldsymbol{\lambda}_{opt}$ cannot be directly computed in most cases, Expectation Propagation attempts to approach it by exploiting the structure of the posterior to construct a sequence of simpler problems.

EP assumes that the distribution of interest decomposes into a product of densities:

$$p(\boldsymbol{\theta}) = Z^{-1} \prod g_i(\boldsymbol{\theta})$$

This is naturally the case when $p(\boldsymbol{\theta})$ is a posterior distribution for independent data, in which case:

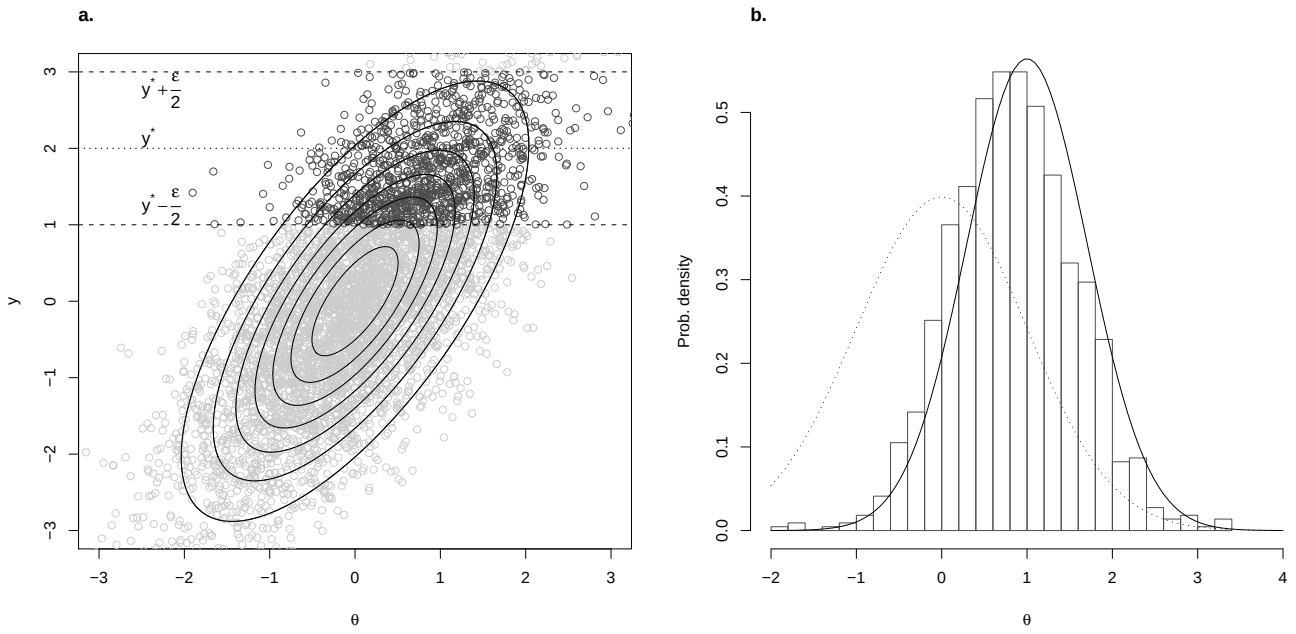

Figure 1. The ABC approximation, and the ABC rejection algorithm. Let $y|\theta \sim \mathcal{N}(\theta, 1)$ and $\theta \sim \mathcal{N}(0, 1)$. Then sampling first from $p(\theta)$ and then from $p(y|\theta)$ will produce samples from $p(y; \theta)$, the contour of which are shown in (**a**) along with some samples. Suppose we observe $y^\star = 2$ and wish to sample from $p(\theta|y^\star)$. The ABC-rejection algorithm accepts all samples from $p(y; \theta)$ that lie in the region defined by $y \in [y^\star - \frac{\epsilon}{2}; y^\star + \frac{\epsilon}{2}]$. We highlight the accepted samples for $\epsilon = 1$. **b.** A histogram of the accepted samples of θ, compared to the exact posterior (solid line) and the prior (dotted line). The mean of the approximation lies between that of the exact posterior and that of the prior, and so does its variance. The approximation is equivalent to knowing only that y is between 1 and 3, instead of knowing its exact value, which by necessity makes the posterior distribution more vague.

$$p(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta}) \prod_{i=1}^{m} g_i(\boldsymbol{\theta}) \tag{2}$$

Here $p_0(\boldsymbol{\theta})$ is the prior distribution, and the g_i's are the likelihoods of each of the m datapoints. Expectation Propagation uses an approximating distribution with a similar structure:

$$q(\boldsymbol{\theta}) \propto p_0(\boldsymbol{\theta}) \prod f_i(\boldsymbol{\theta}) \tag{3}$$

In Expectation Propagation the f_i's are known as the "sites". Each site is a (not necessarily normalised) member of the exponential family : $f_i(\boldsymbol{\theta}) \propto \exp\left(\boldsymbol{\lambda}_i^{\mathbf{t}} \boldsymbol{t}(\boldsymbol{\theta})\right)$, so that:

$$\prod_i f_i(\boldsymbol{\theta}) = \exp\left(\sum \boldsymbol{\lambda}_i^{\mathbf{t}} \boldsymbol{t}(\boldsymbol{\theta})\right) = \exp\left(\boldsymbol{\lambda}^{\mathbf{t}} \boldsymbol{t}(\boldsymbol{\theta})\right) \tag{4}$$

The $\boldsymbol{\lambda}_i$'s are the *site parameters* of the approximation, and $\boldsymbol{\lambda} = \sum \boldsymbol{\lambda}_i$ represents the *global parameters*. The algorithm will update the sites one-by-one, improving the approximation over time.

This is achieved by creating hybrid distributions, which interpolate between the current approximation and the true density. The hybrid is obtained by swapping one site of the approximation with the current site of the true density:

$$h_i(\boldsymbol{\theta}; q, p) = g_i(\boldsymbol{\theta}) \prod_{j \neq i} f_j(\boldsymbol{\theta}) = g_i(\boldsymbol{\theta}) q_{-i}(\boldsymbol{\theta}) \tag{5}$$

The moments of the hybrid are computed, and $\mathbf{Q}, \mathbf{r}$ are updated such that the moments of q match the moments of the hybrid. The site parameters for site i are then updated accordingly. The algorithm loops over sites until convergence. We summarise it as algorithm 1. A more complete description of EP in the exponential family framework can be found in Seeger (2005).

Algorithm 1 Generic EP for exponential families.

Input: a target density $p(\boldsymbol{\theta}) = p_0(\boldsymbol{\theta}) \prod_{i=1}^{n} g_i(\boldsymbol{\theta})$. Initialise $\boldsymbol{\lambda}_0$ to the exponential parameters of the prior p_0, and local site parameters $\boldsymbol{\lambda}_1 \dots \boldsymbol{\lambda}_n = 0$. Set global approximation parameter $\boldsymbol{\lambda}$ to $\boldsymbol{\lambda} = \sum_{i=0}^{n} \boldsymbol{\lambda}_i = \boldsymbol{\lambda}_0$. Pick a site $i \in 1 \dots n$ and loop until convergence:

1. Create hybrid distribution $g_i(\boldsymbol{\theta}) q_{-i}(\boldsymbol{\theta})$ by setting $q_i(\boldsymbol{\theta}) \propto \exp\left(\boldsymbol{\lambda}_{-i}^t \mathbf{t}(\boldsymbol{\theta})\right)$ with $\boldsymbol{\lambda}_{-i} = \boldsymbol{\lambda} - \boldsymbol{\lambda}_i$.
2. Compute moments $\boldsymbol{\eta}_h$ of hybrid distribution, transform to exponential parameters $\boldsymbol{\lambda}_h$.
3. Update site i by setting $\boldsymbol{\lambda}_i = \boldsymbol{\lambda}_h - \boldsymbol{\lambda}_{-i}$, then reset global parameter $\boldsymbol{\lambda}$ to $\boldsymbol{\lambda} = \boldsymbol{\lambda}_h$.

Return moment parameters $\boldsymbol{\eta} = \boldsymbol{\eta}(\boldsymbol{\lambda})$.

3.2. Adapting EP for likelihood-free inference

Assuming that the data vector may be decomposed as $\boldsymbol{y} = (y_1, \dots, y_n)$, an arguably far better target for an likelihood-free inference method would be:

$$p_\epsilon(\boldsymbol{\theta}|\mathbf{y}^\star) \propto \int p_0(\boldsymbol{\theta})\, p_l(\mathbf{y}|\boldsymbol{\theta}) \prod_{i=1}^{n} k_\epsilon(y_i - y_i^\star)\, \mathrm{d}\mathbf{y} \quad (6)$$

which is the ABC posterior without the approximation due to summary statistics, and therefore has the advantage of being exact in the $\epsilon \to 0$ limit. We have seen that this target is not realistic for current ABC algorithms, except in a few special cases (see section 5). However, if the likelihood factorise over datapoints, it becomes a potential target for Expectation Propagation.

In essence, EP requires that one computes the moments of a pseudo-posterior distribution made up of a Gaussian prior and the likelihood for one datapoint. The central idea of ABC-EP is that this can be done efficiently in the ABC context because the likelihood for one datapoint is approximable by:

$$\int_{y_i^\star - \epsilon/2}^{y_i^\star + \epsilon/2} p(y_i|\boldsymbol{\theta}) \mathrm{d}y_i$$

so that the hybrid distribution in equation 5 becomes:

$$h(\boldsymbol{\theta}) \propto \int_{y_i^\star - \epsilon/2}^{y_i^\star + \epsilon/2} p(y_i|\boldsymbol{\theta}) q_{-i}(\boldsymbol{\theta})\, \mathrm{d}y_i \quad (7)$$

In practice the relevant quantities are estimated by simulating $\boldsymbol{\theta}$ values from q_{-i}, then y_i values from $p(y_i|\boldsymbol{\theta})$ and rejecting points too far from y_i^*. The formulas are summarised as Algorithm 2 and Figure 1 provides an illustration. The reason this is efficient is that, since we are only integrating one datapoint at a time, the rejection rate is likely to be tolerably small even for small windows.

Therefore the simplest version of the ABC-EP algorithm is as follows: choose a window size ϵ, then run the standard Expectation Propagation algorithm (algorithm 1), computing the relevant moments of the hybrid distribution using the method described in algorithm 2.

In order for ABC-EP to be effective in practice, we need to limit the number of times the model is simulated. On the other hand, if too few samples are used in the computation of the moments, then EP will become unstable and never converge to a good approximation. Our solution is to sample values of $\boldsymbol{\theta}$ using a Quasi-Monte Carlo strategy (namely a Halton sequence, Lemieux, 2009)) to reduce the variance of moment computations. QMC integration has (worst case) $\mathcal{O}\left(\log(N)^d/N\right)$ convergence, with N the number of samples and d the dimensionality. For the low dimensional problems we are interested in, this is superior to the $\mathcal{O}\left(N^{-1/2}\right)$ convergence of Monte Carlo techniques.

Algorithm 2 Computing the moments of the hybrid distribution in the likelihood-free setting, basic algorithm.

Inputs: a "cavity" distribution $q_{-i}(\boldsymbol{\theta})$ with parameter $\boldsymbol{\lambda}_{-i}$. The true datapoint y_i^*. A window size ϵ.

1. Compute a Quasi-Monte Carlo sequence $\boldsymbol{\theta}^1 \dots \boldsymbol{\theta}^k$ with marginal distribution $q_{-i}(\boldsymbol{\theta})$. Each sample $\boldsymbol{\theta}^m$ in the sequence is obtained via the transformation $\boldsymbol{\theta}^m = \mathbf{U}^\mathbf{t} \mathbf{z}^m + \mathbf{m}$, where $\mathbf{z}^1 \dots \mathbf{z}^n$ is the Halton sequence, $\mathbf{U}$ is the Cholesky factor of the covariance of q_i and $\mathbf{m}$ is its mean.
2. Sample pseudo-datapoints $y_i^1 \dots y_i^k$ using by simulating the model for datapoint i, conditional on parameter values $\boldsymbol{\theta}^1 \dots \boldsymbol{\theta}^k$. Reject all samples for which $||y_i^* - y_i^k|| > \epsilon$. This yields a set of accepted pairs $\left\{\left(\boldsymbol{\theta}_{acc}^1, y_{acc}^1\right), \dots, \left(\boldsymbol{\theta}_{acc}^a, y_{acc}^a\right)\right\}$ of size a.
3. Compute the normalising constant $z_h = \epsilon^{-1} a/k$, and moments $\boldsymbol{\mu}_h = \frac{1}{a} \sum \boldsymbol{\theta}_{acc}^j$ and $\boldsymbol{\Sigma}_h = \frac{1}{a} \sum \boldsymbol{\theta}_{acc}^j \left(\boldsymbol{\theta}_{acc}^j\right)^t - \boldsymbol{\mu}_h \boldsymbol{\mu}_h^\mathbf{t}$.

Return z_h, $\boldsymbol{\mu}_h$, $\boldsymbol{\Sigma}_h$.

4. Toy example: regression under Gaussian errors

We demonstrate some features of the algorithm in the case of regression under Gaussian noise, which of course no one in their right mind would use approximate methods for, but is quite useful as a way to illustrate the comparative properties of ABC-EP and of likelihood-free MCMC samplers.

The model is the classical linear-Gaussian model, with $m = 4$ parameters and $n = 100$ datapoints.

$$\begin{aligned} \mathbf{y} &= \mathbf{X}\mathbf{w} + \mathbf{n} \\ \mathbf{n} &\sim \mathcal{N}\left(0, \sigma^2 \mathbf{I}\right) \end{aligned}$$

We take the model parameters to be the regression weights $\boldsymbol{\theta} = \mathbf{w}$, the prior to be $\boldsymbol{\theta} \sim \mathcal{N}(0, \mathbf{I})$. We chose random uniform values over [0,1] for X_{ij}, and the true values for $\boldsymbol{\theta}$ were picked from a $\mathcal{N}(0, 1)$ distribution. The exact posterior distribution is another Gaussian:

$$\begin{aligned} \boldsymbol{\theta}|\mathbf{y}^\star &\sim \mathcal{N}\left(\sigma^{-2}\mathbf{H}^{-1}\mathbf{X}^\mathbf{t}\mathbf{y}^\star; \mathbf{H}^{-1}\right) \\ \mathbf{H} &= \sigma^{-2}\mathbf{X}^\mathbf{t}\mathbf{X} + \mathbf{I} \end{aligned} \tag{8}$$

We ran ABC-EP for two complete passes over the data, computing the moment of the hybrid distributions using algorithm 2 setting a goal of a minimum of k samples from the truncated distribution. We vary k to vary precision (10 values logarithmically distributed between 10^2 and $10^{4.5}$). ϵ was set to σ.

It is difficult to set up a fair comparison to other likelihood-free MCMC algorithms, chiefly because tuning these algorithms is notoriously difficult and time-consuming (Marin et al., 2011). Effectively, one must choose summary statistics, a proposal distribution, an acceptance parameter and a starting point. The latter point is especially significant: a bad starting point will lead one to choose an extremely high value for ϵ, which will render the MCMC run effectively useless.

We therefore gave MCMC-ABC an unrealistic advantage: minimal sufficient statistics, optimal scaling and optimal starting point. The sufficient statistics were the maximum likelihood estimate for data $\mathbf{y}$ for the true model: $\mathbf{s}(\mathbf{y}) = \left(\mathbf{X}^\mathbf{t}\mathbf{X}\right)^{-1}\mathbf{X}^\mathbf{t}\mathbf{y}$. We used the optimal scaling for the exact posterior, with a Gaussian proposal distribution with variance $(2.38)^2\,\mathbf{H}^{-1}/m$ (Roberts & Rosenthal, 2001). We initialised the chain at the posterior mode $\mathbf{H}^{-1}\mathbf{X}^\mathbf{t}\mathbf{y}^\star$. For the ABC posterior to have a variance comparable to that of the EP posterior it was necessary to set $\epsilon = \sigma/5$. We ran the algorithm for a variable number of iterations (10 values logarithmically distributed between $10^{4.5}$ and 10^6). Because of the high autocorrelation in the chains, we trimmed the values to keep only one iteration in 100. We next discarded the first 100 values as burn-in, and estimated the posterior mean from the mean of the samples. Results are plotted on Figure 2.

5. Example: stochastic predator-prey model

The most prominent advantage of ABC-EP is to be free of summary statistics, targeting directly the truncated posterior of equation 6. In certain special cases, such as the predator-prey model studied here, there are MCMC algorithms that target this posterior as well. We show that ABC-EP remains attractive because it is considerably faster than its MCMC counterparts.

The stochastic predator-prey model (Gillespie, 1977; Boys et al., 2008; Toni et al., 2009) is a nondeterministic version of the classical Lotka-Volterra model of predator-prey dynamics. It is an example of a wider class of models for discrete, interacting populations that is heavily used in chemistry. The model describes the evolution of two discrete populations $x_A(t)$ and $x_B(t)$, in the case when the B's eat the A's. Three type of events can occur: a prey is born (x_A is increased by one), a predator dies (x_B is reduced by 1), or a predator eats a prey (x_A is decreased by 1, x_B is increased by one). The instantaneous probability of these events is given by the following rate equations:

1. Predators die with rate $r_1 x_A(t)$
2. Preys are born with rate $r_2 x_B(t)$
3. A prey is consumed by a predator with rate $r_3 x_A(t) x_B(t)$

We observe the two populations at time $t_1 ... t_n$, and we wish to infer the vector of log-rates $\boldsymbol{\theta} = (\log r_1, \log r_2, \log r_3)$.

Two MCMC samplers are available. When relatively few datapoints have been observed (n small), one can follow the strategy of Toni et al. (2009) and use regular ABC algorithms. A more sophisticated approach can be found in Holenstein (2009). Although presented within the Particle MCMC framework (Andrieu et al., 2010), it can be seen to target an ABC posterior distribution. The ABC posterior is interpreted as the posterior of a state-space model, and a particle filter is used to evaluate the likelihood within a Metropolis-Hastings sampler.

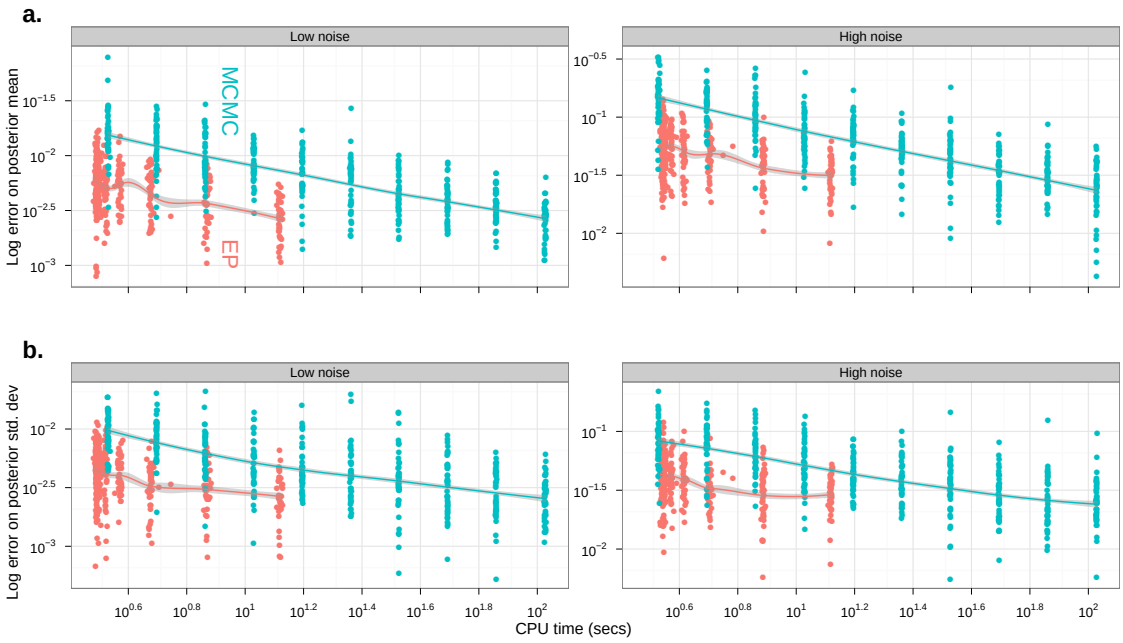

Figure 2. Performance against computational effort for ABC-EP and MCMC-ABC in a toy linear regression example. Both algorithms were implemented in MATLAB and ran on the same machine (Lenovo X201). Running time was measured using the "tic/toc" functions. **a.** Error in estimation of posterior mean (defined as the Euclidean distance of the approximate posterior mean computed by the algorithm, to the exact posterior mean given by equation 8). Each datapoint represents one run of the algorithm: ABC-EP appears in red, and MCMC-ABC in blue. The continuous curves are local smoothers obtained using the R function *loess*. b. Error in estimation of marginal posterior standard deviations of the parameters (defined similarly as the Euclidean distance of the approximate posterior standard deviations to the exact values8).

The Markov property of the underlying model can be used to speed up ABC-EP considerably, by targeting the following approximate posterior:

$$p_\epsilon(\boldsymbol{\theta}) \quad = \quad p_0(\boldsymbol{\theta}) \prod_{t=1}^{n} \int p_l(y_t|y^*_{t-1}, \boldsymbol{\theta}) k_\epsilon(y_t - y^\star_t)\, \mathrm{d}y_t.$$

This approximate posterior is exact in the $\epsilon \to 0$ limit, and decomposes into a product over sites.

Since each of the sites only depends on $p(y_t|y^\star_{t-1}, \boldsymbol{\theta})$, each ABC-EP update only requires simulating the model between two successive sampling times, which can be done relatively cheaply.

To compare the various methods, we simulated the model for $1 \leq t \leq 50$ with true rates $r_1 = 0.4$, $r_2 = 0.01$, $r_3 = 0.3$, to obtain the data plotted in Figure 3. We set a prior with mean $(0, -5, 0)$ and unit variance. We ran ABC-EP with a target sample size of 4000 and an value of 3. On a standard desktop machine this takes 150 seconds. We ran the Particle MCMC algorithm with 1000 particles, 1.7×10^5 iterations and the same value for ϵ, so that the target posterior is identical. The simulation takes two days, and, even when keeping only one particle in 10, the resulting chain showed considerable auto-correlation (CODA estimates an effective sample size of around 350 samples). MCMC-ABC performs even worse - obtaining a decent acceptance rate requires taking ϵ very large, and the resulting posterior distribution is twice as spread as that obtained using Particle MCMC (results not shown). The posterior marginals obtained using MCMC and ABC-EP are shown on Figure 4 - EP-ABC yields essentially identical results in a fraction of the time.

6. Discussion

In the examples we have studied, ABC-EP provides accurate results in a fraction of the time required by other methods, and frees the user from having to choose summary statistics for the dataset. This convenience comes at a cost: ABC-EP requires for the likelihood to factorise, and for the target posterior distribution to be relatively well-behaved. This is of course more difficult to check for models with intractable like-

lihood functions. In practice some care has to be taken in choosing an appropriate parameterisation (avoiding heavy posterior tails for example), and making sure that parameters are not redundant (which could lead to a multimodal posterior). A potential strategy for troublesome posteriors is to treat a subset of the parameters as hyperparameters, running ABC-EP conditional on each value of the hyperparameters in a grid. Another way is to make use of the corrections described in Paquet et al. (2009). Future work will address this issue.

While the current focus of research in Expectation Propagation is on issues of large scale inference (with large number of parameters and/or sites, e.g. Qi et al., 2010 or Seeger et al., 2007), likelihood-free inference is only realistic when the number of parameters is small. The efficiency of our method hangs nearly entirely on how fast and reliably the computation of the moments can be carried out. The simple rejection method used here may be worth replacing with something more sophisticated when simulating the model is relatively slow: an entire array of potential improvements is available. These range from changing the kernel (here uniform), to using non-parametric regression to model the conditional distribution of y_i on $\boldsymbol{\theta}$, then integrating analytically.

On a theoretical level, work is needed to understand the properties of EP, and how these are affected in ABC-EP by the Monte Carlo error incurred in computing the updates. Although EP works almost unreasonably well on certain problems (e.g. Gaussian Process classification, Nickisch & Rasmussen, 2008), nearly recovering the exact mean of the target density, it still lacks strong theoretical support. Some progress has recently been made in this direction, with asymptotic consistency results in special cases (Titterington, forthcoming), but much remains to be done.

Acknowledgements

The authors would like to thank the reviewers for offering valuable comments on the manuscript. This work has benefited from funding from the Bernstein Center for Computational Neuroscience to Simon Barthelmé. Nicolas Chopin acknowledges support from the ANR grant ANR-008-BLAN-0218 "BigMC" of the French Ministry of Research.

References

Andrieu, C., Doucet, A., and Holenstein, R. Particle Markov chain Monte Carlo methods. *J. R. Statist. Soc. B*, 72(3):269–342, 2010. doi: 10.1111/j.1467-9868.2009.00736.x.

Beaumont, M.A., Zhang, W., and Balding, D.J. Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025, 2002.

Beaumont, Mark A. Approximate Bayesian Computation in Evolution and Ecology. *Annual Review of Ecology, Evolution, and Systematics*, 41(1):379–406, 2010. doi: 10.1146/annurev-ecolsys-102209-144621.

Blum, M. G. B. Approximate Bayesian Computation: A Nonparametric Perspective. *J. Am. Statist. Assoc.*, 105(491):1178–1187, 2010.

Blum, Michael and François, Olivier. Non-linear regression models for Approximate Bayesian Computation. *Statist. Comput.*, 20(1):63–73, January 2010. ISSN 0960-3174. doi: 10.1007/s11222-009-9116-0.

Boys, R.J., Wilkinson, D.J., and Kirkwood, T.B.L. Bayesian inference for a discretely observed stochastic kinetic model. *Statist. Comput.*, 18(2):125–135, 2008. ISSN 0960-3174. doi: 10.1007/s11222-007-9043-x.

Del Moral, P., Doucet, A., and Jasra, A. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68 (3):411–436, 2006.

Del Moral, P., Doucet, A., and Jasra, A. An Adaptive Sequential Monte Carlo Method for Approximate Bayesian Computation. Technical report, 2008.

Gillespie, Daniel T. Exact stochastic simulation of coupled chemical reactions. *The Journal of Physical Chemistry*, 81(25):2340–2361, December 1977. doi: 10.1021/j100540a008.

Gouriéroux, C., Monfort, A., and Renault, E. Indirect Inference. *Journal of Applied Econometrics*, 8:S85–S118, 1993. ISSN 08837252. doi: 10.2307/2285076.

Holenstein, R. *Particle Markov Chain Monte Carlo*. PhD thesis, University of British Columbia, 2009.

Lemieux, Christiane. *Monte Carlo and Quasi-Monte Carlo Sampling (Springer Series in Statistics)*. Springer, 1 edition, February 2009. ISBN 0387781641.

Marin, Jean-Michel, Pudlo, Pierre, Robert, Christian P., and Ryder, Robin. Approximate Bayesian Computational methods. 2011.

Marjoram, Paul, Molitor, John, Plagnol, Vincent, and Tavaré, Simon. Markov Chain Monte Carlo without Likelihoods. 100(26):15324–15328, 2003. ISSN 00278424. doi: 10.2307/3149004.

Minka, T.P. Expectation Propagation for approximate Bayesian inference. *PRoceedings of Uncertainty in Artificial Intelligence*, 17:362–369, 2001.

Nickisch, H. and Rasmussen, C.E. Approximations for Binary Gaussian Process Classification. *J. Machine Learning Research*, 9(10):2035–2078, October 2008.

Nunes, Matthew A. and Balding, David J. On optimal selection of summary statistics for approximate Bayesian computation. *Statistical applications in genetics and molecular biology*, 9(1), 2010. ISSN 1544-6115. doi: 10.2202/1544-6115.1576.

Opper, Manfred and Winther, Ole. Expectation Consistent Approximate Inference. *J. Machine Learning Research*, 6:2177–2204, December 2005. ISSN 1532-4435.

Paquet, Ulrich, Winther, Ole, and Opper, Manfred. Perturbation Corrections in Approximate Inference: Mixture Modelling Applications. *J. Machine Learning Research*, 10:1263–1304, June 2009.

Pritchard, J.K., Seielstad, M.T., Perez-Lezaun, A., and Feldman, M.W. Population growth of human Y chromosomes: a study of Y chromosome microsatellites. *Molecular Biology and Evolution*, 16(12):1791, 1999. ISSN 0737-4038.

Qi, Alan, Abdel-Gawad, Ahmed, and Minka, Thomas. Sparse-posterior Gaussian Processes for general likelihoods. In Grünwald, P. and Spirtes, P. (eds.), *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2010.

Roberts, Gareth O. and Rosenthal, Jeffrey S. Optimal Scaling for Various Metropolis-Hastings Algorithms. *Statist. Science*, 16(4):351–367, 2001. ISSN 08834237. doi: 10.2307/3182776.

Seeger, M. Expectation Propagation for Exponential Families. Technical report, Univ. California Berkeley, 2005.

Seeger, Matthias, Gerwinn, Sebastian, and Bethge, Matthias. Bayesian Inference for Sparse Generalized Linear Models. In *ECML '07: Proceedings of the 18th European conference on Machine Learning*, pp. 298–309. Springer-Verlag, 2007. ISBN 978-3-540-74957-8. doi: 10.1007/978-3-540-74958-5_29.

Toni, Tina, Welch, David, Strelkowa, Natalja, Ipsen, Andreas, and Stumpf, Michael P. H. Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of The Royal Society Interface*, 6 (31):187–202, February 2009. ISSN 1742-5689. doi: 10.1098/rsif.2008.0172.

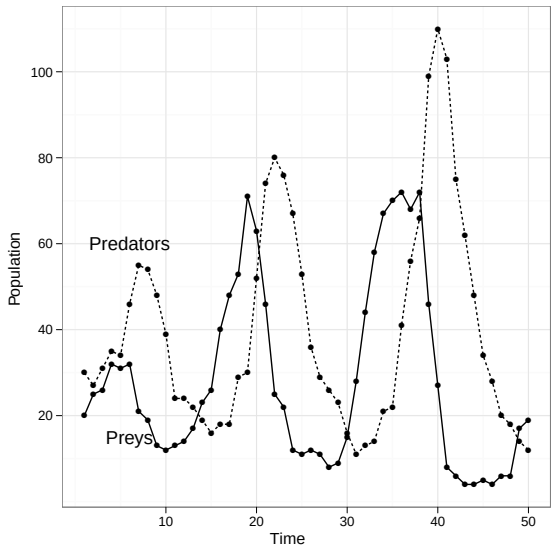

Figure 3. Simulated dataset for the Lotka-Volterra predator-prey model. We plot the evolution of population counts across time, as sampled at discrete time points.

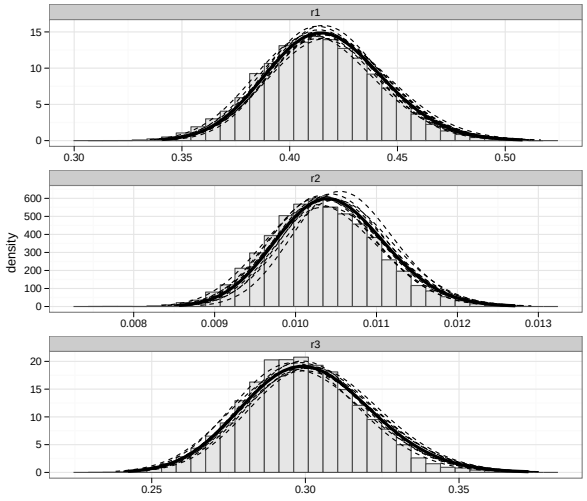

Figure 4. Comparison of the posterior distributions for the rate parameters obtained using EP-ABC and Particle MCMC. The Particle MCMC samples are represented as histograms and correspond to an ϵ value of 3. EP-ABC was ran 10 times. Due to Monte Carlo variance the results are not identical every time, and we plot the corresponding individual densities (dotted lines) as well as that obtained by averaging the parameters over the 10 runs (thick lines).