Piecewise Approximate Bayesian Computation (PW–ABC)

Fast inference for discretely observed Markov models

Theo Kypraios

http://www.maths.nott.ac.uk/~tk

Department of Mathematics and Statistics, Lancaster University
Joint work:

- Simon Preston (University of Nottingham)
- Simon White (Biostatistics Unit, MRC @ Cambridge)

Piecewise Approximate Bayesian Computation: fast inference for discretely observed Markov models using a factorised posterior distribution

S. R. White, T. Kypraios, S. P. Preston

Funding:

The University of Nottingham

THE ROYAL SOCIETY

EPSRC
Engineering and Physical Sciences Research Council
A Motivating Example: Lotka–Volterra

Suppose our data are a set of observations denoted \( \mathcal{X} = \{x_1, \ldots, x_n\} = \{x(t_1), \ldots, x(t_n)\} \) of state variable \( x \in \mathbb{R}^m \) at time points \( t_1, \ldots, t_n \).
Motivation:

1. Given some observed data . . .

2. . . . that we assume to have been generated by a (stochastic) model, for example . . .

\[ Y_1 \xrightarrow{r_1} 2Y_1, \quad Y_1 + Y_2 \xrightarrow{r_3} 2Y_2, \quad Y_2 \xrightarrow{r_3} \emptyset, \quad (1) \]

which respectively represent prey birth, predation and predator death.

3. . . . we wish to make inference for the model parameters \((r_1, r_2, r_3)\).
Motivation:

1. Given some observed data . . .

2. . . . that we assume to have been generated by a (stochastic) model, for example . . .

\[ Y_1 \xrightarrow{r_1} 2 Y_1, \quad Y_1 + Y_2 \xrightarrow{r_3} 2 Y_2, \quad Y_2 \xrightarrow{r_3} \emptyset, \quad (1) \]

which respectively represent prey birth, predation and predator death.

3. . . . we wish to make inference for the model parameters \((r_1, r_2, r_3)\).
Motivation:

1. Given some observed data . . .

2. . . . that we assume to have been generated by a (stochastic) model, for example . . .

\[ Y_1 \xrightarrow{r_1} 2 Y_1, \quad Y_1 + Y_2 \xrightarrow{r_2} 2 Y_2, \quad Y_2 \xrightarrow{r_3} \emptyset, \quad (1) \]

which respectively represent prey birth, predation and predator death.

3. . . . we wish to make inference for the model parameters \((r_1, r_2, r_3)\).
Denote by $X$ our observed data and by $\theta$ the parameter(s) of interest.

If we were to employ a maximum likelihood approach then we must be able to write down/evaluate the likelihood $\pi(X|\theta)$ i.e. the probability of observing the data $X$ (what we have observed) for all parameter values $\theta$ . . .

. . . and then find which parameter(s) $\theta$ maximise the likelihood.

Use asymptotic theory and obtain (approximate) confidence intervals to quantify uncertainty around $\theta$. 
Denote by $\mathcal{X}$ our observed data and by $\theta$ the parameter(s) of interest.

If we were to employ a maximum likelihood approach then we must be able to write down/evaluate the likelihood $\pi(\mathcal{X} | \theta)$ i.e. the probability of observing the data $\mathcal{X}$ (what we have observed) for all parameter values $\theta$ ...

... and then find which parameter(s) $\theta$ maximise the likelihood.

Use asymptotic theory and obtain (approximate) confidence intervals to quantify uncertainty around $\theta$. 
Interested in cases where conventional methods fail simply don’t work, for example:
- intractable likelihood;
- likelihood is costly to compute;
- …

Idea: Explore model dynamics to design efficient designing new methods/algorithms.

For the purposes of this talk we adopt a Bayesian framework.
Suppose we have discrete data $\mathcal{X}$, prior $\pi(\theta)$ for parameter(s) $\theta$; interested in the posterior, $\pi(\theta|\mathcal{X})$.

Consider the following algorithm:

**Algorithm 1**

Exact Bayesian Computation (EBC)

1: Sample $\theta^*$ from $\pi(\theta)$.
2: Accept $\theta^*$ with probability equal to $\pi(\mathcal{X}|\theta^*)$
3: Repeat.

* The accepted values will be (exact) draws from $\pi(\theta|\mathcal{X})$.

* Note that this algorithm requires that we are able to compute the likelihood, $\pi(\mathcal{X}|\theta)$, for any $\theta$ (Step 2).
Consider the following algorithm$^1$:

**Algorithm 2**

**Exact Bayesian Computation (EBC)**

1: Sample $\theta^*$ from $\pi(\theta)$.
2: Simulate dataset $x^*$ from the model using parameters $\theta^*$.
3: Accept $\theta^*$ if $x^* = x$, otherwise reject.
4: Repeat.

* Algorithm 2 is equivalent to Algorithm 1 and evaluating $\pi(X|\theta)$ $\iff$ simulate an event which occurs with that probability.

* That means that the calculation of the likelihood is unnecessary as long as we can simulate from our stochastic model.

Algorithm 2 is only of practical use if $\mathcal{X}$ is discrete, else the acceptance probability in Step 3 is zero.

For continuous distributions Pritchard et al. (1999) suggested the following algorithm:

**Algorithm 3**

Approximate Bayesian Computation (ABC)

As Algorithm 2, but with step 3 replaced by:

3′: Accept $\theta^*$ if $d(s(\mathcal{X}'), s(\mathcal{X}^*)) \leq \varepsilon$, otherwise reject.

where $d(\cdot, \cdot)$ is a distance function, usually taken to be the $L^2$-norm of the difference between its arguments; $s(\cdot)$ is a function of the data; and $\varepsilon$ is a tolerance.
In practice iss rarely possible to use an $s(\cdot)$ which is sufficient, or to take $\varepsilon$ especially small (or zero).

ABC requires a careful choice of $s(\cdot)$ and $\varepsilon$ to make the acceptance rate tolerably large, at the same time as trying not to make the ABC posterior too different from the true posterior, $\pi(\theta|X)$.

Over the last decade, a wide range of extensions to the original ABC algorithm have been developed (MCMC-ABC, SMC-ABC, Semi-Automatic ABC . . .)

. . . , however, computational cost remains a central issue since it determines the balance that can be made between Monte Carlo error/bias (via summary stats).
Interested in exploring cases (i.e. models/data) and methods where ideally, exact Monte-Carlo inference can be drawn in practice without having to compute likelihoods either because it is

- too expensive to compute
- or, intractable;

or, difficult to maximise or sample from the posterior distribution of interest.

If exact inference seems infeasible → efficient, but approximate likelihood-free inference.

A guiding principle is to take every opportunity to exploit model structure to minimize computational costs.
The Markov property enables the likelihood to be written as

$$\pi(\mathcal{X}|\theta) = \pi(x_1, x_2, \ldots, x_n|\theta)$$

$$\pi(\mathcal{X}|\theta) = \pi(x_1|\theta) \left( \prod_{i=2}^{n} \pi(x_i|x_{i-1}, \ldots, x_1, \theta) \right)$$

$$= \pi(x_1|\theta) \left( \prod_{i=2}^{n} \pi(x_i|x_{i-1}, \theta) \right), \quad (2)$$

if $n$ is large then we can ignore the contribution of the first data point ($x_1$) to the likelihood, and write

$$\pi(\mathcal{X}|\theta) = \prod_{i=2}^{n} \pi(x_i|x_{i-1}, \theta)$$
Hence the posterior as

\[ \pi(\theta | \mathcal{X}) \propto \pi(\theta) \cdot \pi(\mathcal{X} | \theta) \]

\[ \propto \pi(\theta) \cdot \prod_{i=2}^{n} \pi(x_i | x_{i-1}, \theta) \]

\[ \propto \pi(\theta) \cdot \prod_{i=2}^{n} \left( \frac{\pi(x_i | x_{i-1}, \theta)\pi(\theta)}{\pi(\theta)} \right) \]

\[ \propto \pi(\theta)^{(2-n)} \prod_{i=2}^{n} \pi(x_i | x_{i-1}, \theta)\pi(\theta) \]

\[ \propto \pi(\theta)^{(2-n)} \prod_{i=2}^{n} \phi_i(\theta). \]

where

\[ \phi_i(\theta) = c_i^{-1} \pi(x_i | x_{i-1})\pi(\theta) \]

\[ c_i = \int \pi(x_i | x_{i-1})\pi(\theta) \, d\theta \text{ [normalising constant]} \]

[don’t have to necessarily use \( \pi(\theta) \) here!]
Essentially, the density of the posterior distribution of interest, $\pi(\theta|X)$, has been decomposed into a product involving densities $\phi_i(\theta)$, each of which depends only on a pair of data points $\{x_{i-1}, x_i\}$:

$$\pi(\theta|X) \propto \pi(\theta)^{(2-n)} \prod_{i=2}^{n} \phi_i(\theta)$$

(3)

where $\phi_i(\theta) = c_i^{-1}\pi(x_i|x_{i-1})\pi(\theta)$.

- If $\pi(x_i|x_{i-1}, \theta)$ is not available/intractable/difficult to compute then so $\phi_i(\theta)$ is and decomposing $\pi(\theta|X)$ will not be of much help.

- However, if we can simulate from each distribution with density $\propto \phi_i(\theta)$, i.e. simulate $x_i|x_{i-1}$, then it turns out that we can recover the posterior density, $\pi(\theta|X)$. 
Although the transition density $\pi(x_i|x_{i-1})$ might be intractable, we can draw samples from each density

$$\phi_i(\theta) \propto \pi(\theta)\pi(x_i|x_{i-1}, \theta), \quad i = 2, \ldots n.$$ using the following algorithm:

**Algorithm 4 : EBC (ABC) within each interval**

1: Sample $\theta^*$ from $\pi(\theta)$.
2: Simulate $x_i^*|x_{i-1}$ from the model using $\theta^*$.
3: Accept $\theta^*$ if $x_i = x_i^*$ (or $d(s(x_i), s(x_i^*)) \leq \varepsilon$), otherwise reject.
4: Repeat.

In other words, apply (independent) EBC/ABC for each pair/interval $(x_i, x_{i-1})$ to draw from each density $\phi_i(\theta)$. 
Algorithm 5 Piece-Wise Approximate Bayesian Computation

for $i = 2$ to $n$ do

a: Apply the ABC Algorithm to draw $m$ approximate (or exact, if $s(\cdot) = \text{Identity}(\cdot)$ and $\varepsilon = 0$) samples from $\tilde{\phi}_i(\theta)$;

b: Using the samples calculate a density estimate, $\hat{\phi}_i(\theta)$, of $\tilde{\phi}_i(\theta)$.

end for

Substitute the density estimates $\hat{\phi}_i(\theta)$ into (19) to calculate an estimate, $\hat{\pi}(\theta|x)$, of $\pi(\theta|x)$. 
The rationale of the piecewise approach is to reduce the dimension for ABC . . .

. . . replacing a high-dimensional problem with multiple low-dimensional ones.

In standard ABC the summary statistic, $s(\cdot)$, is the tool used to reduce the dimension.

In PW-ABC, with dimension already reduced by the factorisation in (3), we can take $s(\cdot) = \text{Identity}(\cdot)$ and typically use a much smaller $\varepsilon$. 
How to Compute the Density Estimates $\hat{\phi}_i(\theta)$?

Recall that the full posterior distribution of the parameters has been re-written as

$$\pi(\theta|\mathcal{X}) \propto \pi(\theta)^{(2-n)} \prod_{i=2}^{n} \phi_i(\theta)$$

However, the question remains of how to calculate the density estimates, $\hat{\phi}_i(\theta)$.

We propose two approaches:

1. using a Gaussian approximation,
2. using a kernel density estimate.
This approach requires a kernel density estimation (KDE) on each \( \phi_i(\theta) \) . . .

. . . and then multiplying the KDEs pointwise adjusting for the \((n - 2)\) prior densities.

In principle this should work . . . and it does work, as long as you are careful and you have a descent number of posterior samples in each interval! [marginal likelihood]
KDEs can be hard to deal with; especially in high dimensions!

Alternatively, we could approximate each $\phi_i(\theta)$ with a (multivariate) Gaussian distribution

$$\hat{\phi}_i(\theta) = \text{MVN}(\mu_i, \Sigma_i)$$

where $\mu_i$ and $\Sigma_i$ could be the sample mean and the sample variance-covariance matrix;

Take advantage of the appealing property that the product

$$\prod_{i=2}^{n} \hat{\phi}_i(\theta)$$

leads to another Gaussian density too . . . ,

. . . which combined with $(n - 2)$ (Gaussian) prior densities leads, finally, to a Gaussian approximation to the full posterior density $\pi(\theta|X)$.

As a by-product we get the marginal likelihood $\pi(X)$ which can be used for model choice.
KDEs are known to perform poorly on bounded supports → transform the parameters ($\theta$).

**Which Kernel to use?**

We follow Fukunaga (1972) “sphering approach” which selects the bandwidth so that the shape of the kernel mimics the shape of the sample;

**Easy to select** an “optimal” bandwidth when doing KDE in each interval, but not so easy when looking at the product of KDEs.

**The Gaussian approximation to each** $\phi_i(\theta)$ may not be necessarily good and this will lead to biased estimates $\pi(\theta|\mathcal{X})$. 
Applications
Consider the following integer-valued autoregressive model of order 1, known as INAR(1) [Al-Osh and Alzaid, 1987],:

\[ X_t = \alpha \circ X_{t-i} + Z_t, \quad t \in \mathbb{Z}, \]

where \( Z_t \) are i.i.d. integer-valued random variables and assumed to be independent of the \( X_t \).

The operator \( \alpha \circ \) denotes binomial thinning defined by

\[ \alpha \circ W = \begin{cases} 
\text{Binomial}(W, \alpha), & W > 0, \\
0, & W = 0,
\end{cases} \]

This model falls into the class of models that one can take advantage of Piecewise approaches.
We generated 100 observations from an INAR(1) process using parameters \( \theta = (\alpha, \lambda) = (0.7, 1) \) and \( X(0) = 2 \).

We make inference on the transformed parameters \( \tilde{\alpha} = \logit(\alpha) = \log(\alpha) - \log(1 - \alpha) \) and \( \tilde{\lambda} = \log(\lambda) \) . . .

. . . with priors of \( \text{Norm}(0, 3^2) \) on the transformed parameters.

For the EBC algorithm (on the whole dataset) the probability of acceptance is around \( 10^{-100} \), which is prohibitively small.

Even the ABC algorithm requires a value of \( \epsilon \) so large that sequential approaches are needed, e.g. SMC-ABC, [Toni et al., 2009].
The INAR Dataset
PWEBC on INAR Models
(Gaussian)PW-EBC Does Not Seem to Work
The CIR model is a stochastic differential equation (SDE) describing evolution of an interest rate, $X(t)$:

$$dX(t) = a(b - X(t))dt + \sigma \sqrt{X(t)}dW(t),$$

where $a$, $b$ and $\sigma$ respectively determine the reversion speed, long-run value and volatility, and $W(t)$ denotes a standard Brownian motion.

The density of $X(t_i)|X(t_j)$, $a$, $b$, $\sigma$ $(t_i > t_j)$ is a non-central chi-square and hence the likelihood is known in closed form.

This example allows to illustrate the use of PW-ABC in the context of continuous data.
We generated $n = 10$ equally spaced observations from a CIR process with parameters $(a, b, \sigma) = (0.5, 1, 0.15)$ and $X(0) = 1$ on the interval $t \in [0, 4.5]$.

Treating $a$ and $\sigma$ as known, we performed inference on the transformed parameter $\theta = \log(b)$ with a Uniform prior on the interval $(-5, 2)$.

Using $\varepsilon = 10^{-2}$ we drew samples of size $m = 10,000$ for each $\varphi_i(\theta), \ i = 1, \ldots, 9$, achieving acceptance rates around 1.5% on average.
The Figure below shows how the posterior density targeted by PW-ABC depends on $\varepsilon$, and how it converges to the true posterior density as $\varepsilon \to 0$. 
The stochastic(LV) model is a model of predator–prey dynamics.

Let $Y_1$ and $Y_2$ denote the number of prey and predators respectively, and suppose $Y_1$ and $Y_2$ are subject to the following reactions

$$Y_1 \xrightarrow{r_1} 2Y_1, \quad Y_1 + Y_2 \xrightarrow{r_2} 2Y_2, \quad Y_2 \xrightarrow{r_3} \emptyset,$$

which respectively represent prey birth, predation and predator death.

We wish to make inference for vector of rates $\mathbf{r} = (r_1, r_2, r_3)$. 
Likelihood-Based Inference for the LV model

- Inference is simple if the type and precise time of each reaction is observed.

- However, a more common setting is where the population sizes are only observed at discrete time points → likelihood is not available.

- Reversible-Jump MCMC has been developed in this context [e.g. Boys et al., 2008] but require considerable expertise to implement.

- Other approaches include model approximations using SDEs (e.g. Golightly and Wilkinson 2006, 2007) and more recently, Particle MCMC (Wilkinson, 2012)

- On the other hand, simulating realizations from this model is straightforward (e.g. using the Gillespie algorithm).
PW-EBC vs Particle MCMC (sd=2)
Conclusions – Future Work

- PW–ABC takes advantage of the model’s dynamics → inference.

- If $\pi(\theta)$ is too uninformative then PW–ABC will suffer from the same problems as (standard) ABC → use SMC–ABC within each interval.

- PW–SMC–ABC?

- Use a mixture of Gaussians – For efficiency, use some sort of sparsity–induced priors.

- Scope for the theoretical development on the choice of bandwidth for products of KDES.

Suppose that $x$ follow as a distribution whose density

$$p(x) = \prod_{i=1}^{n} p_i(x)$$

where

- we know how to sample from each $p_i(x), \quad i = 1, \ldots, n$
- but we don’t have an explicit expression for $p_i(x)$. 

Is it possible to (efficiently) draw samples from $p(x)$?