Coupled Markov chains with applications to Approximate Bayesian Computation for model based clustering

E. Bertoni, M. Caldarini, F. Di Filippo, G. Gabrielli, E. Musiari

11 november 2021

A complex problem

Our goal is to propose a method which solves these two problems:

- find a way to speed up computation time by parallelizing a Monte Carlo Markov chain method;
- tackle the case in which the likelihood function has a computationally intractable evaluation.

A complex problem

Our goal is to propose a method which solves these two problems:

- find a way to speed up computation time by parallelizing a Monte Carlo Markov chain method;
  $\Longrightarrow$ Unbiased Markov chain Monte Carlo methods with couplings;
- tackle the case in which the likelihood function has a computationally intractable evaluation.

A complex problem

Our goal is to propose a method which solves these two problems:

- find a way to speed up computation time by parallelizing a Monte Carlo Markov chain method;
  $\Longrightarrow$ Unbiased Markov chain Monte Carlo methods with couplings;
- tackle the case in which the likelihood function has a computationally intractable evaluation.
  $\Longrightarrow$ Approximate Bayesian Computation

Unbiased Markov chain Monte Carlo methods with couplings

E. Bertoni, M. Caldarini, F. Di Filippo, G. Gabrielli, E. Musiari

The road to parallelization: coupling of Markov chains

In order to parallelize we need an **unbiased** estimator. Standard Markov

chain Monte Carlo methods are potentially biased for any fixed number of iterations.

P. Glynn and C. Rhee proposed in 2018 the class of exact estimations algorithms using coupling of Markov chain.

The road to parallelization: coupling of Markov chains

In order to parallelize we need an **unbiased** estimator. Standard Markov chain Monte Carlo methods are potentially biased for any fixed number of iterations.

P. Glynn and C. Rhee proposed in 2018 the class of exact estimations algorithms using coupling of Markov chain.

What is the coupling of Markov chains? The coupling of two probability distributions μ and ν refers to the construction of a bivariate probability distribution whose marginals are the original distributions μ and ν. Markov chain coupling allow to **reduce the convergence time**.

Rhee–Glynn estimator I

The goal is to estimate

$$\mathbb{E}_\pi[h(X)] = \int h(x)\pi(\mathrm{d}x).$$

The estimator we are going to construct is based on a coupled pair of Markov chains, $(X_t)_{t\geq 0}$ and $(Y_t)_{t\geq}$, which marginally start from π_0 and evolve accordingly to P.

We consider some assumptions:

1. as $t \to \infty$,

   $$\mathbb{E}[h(X_t)] \to \mathbb{E}_\pi[h(X)];$$

   and there exists $\eta > 0$ and $D < \infty$ such that $\mathbb{E}[|h(X_t)|^{2+\eta}] \leq D$ for all $t \geq 0$;

We consider some assumptions:

1. as $t \to \infty$,
$$\mathbb{E}[h(X_t)] \to \mathbb{E}_\pi[h(X)];$$
and there exists $\eta > 0$ and $D < \infty$ such that $\mathbb{E}[|h(X_t)|^{2+\eta}] \leq D$ for all $t \geq 0$;
2. the chains are such that the meeting time
$$\tau = \inf\{t \geq 1 : X_t = Y_{t-1}\}$$
satisfies $\mathbb{P}(\tau > t) \leq C\delta^t$ for all $t \geq 0$, for some constants $C < \infty$ and $\delta \in (0, 1)$;

We consider some assumptions:

1. as $t \to \infty$,
$$\mathbb{E}[h(X_t)] \to \mathbb{E}_\pi[h(X)];$$
and there exists $\eta > 0$ and $D < \infty$ such that $\mathbb{E}[|h(X_t)|^{2+\eta}] \leq D$ for all $t \geq 0$;
2. the chains are such that the meeting time
$$\tau = \inf\{t \geq 1 : X_t = Y_{t-1}\}$$
satisfies $\mathbb{P}(\tau > t) \leq C\delta^t$ for all $t \geq 0$, for some constants $C < \infty$ and $\delta \in (0, 1)$;
3. the chains stay together after meeting:
$$X_t = Y_{t-1} \text{ for all } t \geq \tau.$$

Rhee–Glynn estimator II

Thanks to the previous assumptions we can prove that:

$$\mathbb{E}_\pi[h(X)] = \mathbb{E}[h(X_k) + \sum_{t=k+1}^{\tau-1} \{h(X_t) - h(Y_{t-1})\}];$$

and we define the Rhee–Glynn estimator as:

$$H_k(X, Y) = h(X_k) + \sum_{t=k+1}^{\tau-1} \{h(X_t) - h(Y_{t-1})\}$$

which is **unbiased** by construction.

Time-averaged estimator

In such form, Rhee-Glynn is not computationally feasible, the time-averaged estimator keeps the assumptions allowing the computation:

$$H_{k:m} = \frac{1}{m-k+1} \sum_{l=k}^{m} h(X_l) + \sum_{l=k+1}^{\tau-1} \min(1, \frac{l-k}{m-k+1}) \{h(X_l) - h(Y_{l-1})\}$$

Time-averaged estimator

In such form, Rhee-Glynn is not computationally feasible, the time-averaged estimator keeps the assumptions allowing the computation:

$$H_{k:m} = \underbrace{\frac{1}{m-k+1}\sum_{l=k}^{m} h(X_l)}_{MCMC_{k:m}} + \underbrace{\sum_{l=k+1}^{\tau-1} \min(1, \frac{l-k}{m-k+1})\{h(X_l) - h(Y_{l-1})\}}_{BC_{k:m}}$$

- $MCMC_{k:m}$ is the standard MCMC average;
- $BC_{k:m}$ is the bias correction;

Time-averaged estimator

In such form, Rhee-Glynn is not computationally feasible, the time-averaged estimator keeps the assumptions allowing the computation:

$$H_{k:m} = \underbrace{\frac{1}{m-k+1}\sum_{l=k}^{m} h(X_l)}_{MCMC_{k:m}} + \underbrace{\sum_{l=k+1}^{\tau-1} \min(1, \frac{l-k}{m-k+1})\{h(X_l) - h(Y_{l-1})\}}_{BC_{k:m}}$$

- $MCMC_{k:m}$ is the standard MCMC average;
- $BC_{k:m}$ is the bias correction;

- $k-1$ number of burn-in iterations;
- m is a fixed integer of number of maximum iterations;
- τ is a random variable representing the meeting time.

Application to Metropolis-Hasting algorithm I

The MH **algorithm** adapted with couplings:

1. draw X_0 and Y_0 from an initial distribution π_0 and draw $X_1 \sim P(X_0, \cdot)$;
2. set $t = 1$: while $t < \max\{m, \tau\}$ and:
   - a draw $(X_{t+1}, Y_t) \sim \bar{P}\{(X_t, Y_{t-1}), \cdot\}$;
   - b set $t \leftarrow t + 1$;
3. compute

$$H_{k:m}(X, Y)$$

with the time-averaged estimator.

Application to Metropolis-Hasting algorithm II

The following is the **algorithm** to calculate the coupled kernel $\bar{P}\{(X_t, Y_{t-1}), \cdot\}$ via MH:

1. sample $(X^\star, Y^\star)|(X_t, Y_{t-1})$ from a maximal coupling of $q(X_t, \cdot)$ and $q(Y_{t-1}, \cdot)$;
2. sample $U \sim \mathcal{U}([0,1])$;
3. if
$$U \leq \min\left\{1, \frac{\pi(X^\star)q(X^\star, X_t)}{\pi(X_t)q(X_t, X^\star)}\right\}$$
then $X_{t+1} = X^\star$; otherwise $X_t = X_{t-1}$;
4. if
$$U \leq \min\left\{1, \frac{\pi(Y^\star)q(Y^\star, Y_t)}{\pi(Y_t)q(Y_t, Y^\star)}\right\}$$
then $Y_{t+1} = Y^\star$; otherwise $Y_t = Y_{t-1}$.

Approximate Bayesian Computation

Likelihood-free methods I

To solve this issue we can use methods based on the **approximation of the likelihood function**, called *Likelihood-free methods*. Here the **algorithm**: *Inputs:*

- a target posterior density $\pi(\theta|y_{obs}) \propto p(y_{obs}|\theta)\pi(\theta)$, consisting of a prior distribution $\pi(\theta)$ and a procedure of generating data under the model $p(y_{obs}|\theta)$;
- a proposal density $g(\theta)$, with $g(\theta) > 0$ if $\pi(\theta|y_{obs}) > 0$;
- an integer $N > 0$.

Likelihood-free methods I

Sampling for $i = 1, ..., N$:

1. generate $\theta^{(i)} \sim g(\theta)$ from sampling density g;
2. generate $y \sim p(y|\theta^{(i)})$ from the likelihood;
3. if $y = y_{obs}$, then accept $\theta^{(i)}$ with probability $\frac{\pi(\theta^{(i)})}{Kg(\theta^{(i)})}$, where $K \geq \max_\theta \frac{\pi(\theta)}{g(\theta)}$; else go to 1.

Output:

- a set of parameter vectors $\theta^{(1)}, ..., \theta^{(N)}$ which are samples from $\pi(\theta|y_{obs})$.

Likelihood-free methods II

Is this an efficient method for complex analysis?

Likelihood-free methods II

Is this an efficient method for complex analysis?

3 If $\| y - y_{obs} \| \leq h$, then accept $\theta^{(i)}$ with probability $\frac{\pi(\theta^{(i)})}{Kg(\theta^{(i)})}$, where $K \geq \max_\theta \frac{\pi(\theta)}{g(\theta)}$.

Else go to 1.

Approximate Bayesian Computation

We focused on a particular case of the Likelihood-free methods: the *Approximate Bayesian Computation (ABC)*.

The aim: find a practical way of performing Bayesian analysis, while keeping the approximation and the computation to a minimum.

The likelihood-free rejection algorithm is sampling from the joint distribution $\propto \mathbb{I}(\| y - y_{obs} \| \leq h) p(y|\theta)\pi(\theta)$
$\Longrightarrow$ replace the indicator function with a standard smoothing kernel function $K_h(u)$, with $u = \| y - y_{obs} \|$:

$$K_h(u) = \frac{1}{h} K(\frac{u}{h})$$

Hence:

$$\pi_{ABC}(\theta, y|y_{obs}) \propto K_h(u) p(y|\theta)\pi(\theta)$$

Summary statistics I

Is this feasible in practice?

In practice: difficult to have $y \approx y_{obs}$ from $p(y|\theta)$, unless y_{obs} very low dimensional or $p(y|\theta)$ factorises into low-dimensional components.
Thus we should use a large h, obtaining a poor posterior approximation!
$\Longrightarrow$ use summary statistics $s = S(y)$

$$\pi_{ABC}(\theta|s_{obs})$$

Summary statistics II

Critical decision: choice of summary statistics

Dimension of summary statistics:

- large enough to contain as much as information about observed data as possible
- low enough to avoid curse of dimensionality of matching s and s_{obs}

$\Longrightarrow$ choose sufficient statistics, such that:

$$\pi(\theta | s_{obs}) \equiv \pi(\theta | y_{obs})$$

Distance measure

Distance measure: substantial impact on ABC algorithm efficiency

$$\| s - s_{obs} \| = (s - s_{obs})^\top \Sigma^{-1} (s - s_{obs})$$

- $\Sigma =$ identity matrix → Euclidean distance
- $\Sigma =$ diagonal matrix of non-zero weights → Weighted Euclidean distance
- $\Sigma =$ full covariance matrix of s → Mahalanobis distance

ABC rejection sampling Algorithm

The following is an ABC **algorithm**: *Inputs:*

- a target posterior density $\pi(\theta|y_{obs}) \propto p(y_{obs}|\theta)\pi(\theta)$, consisting of a prior distribution $\pi(\theta)$ and a procedure of generating data under the model $p(y_{obs}|\theta)$;
- a proposal density $g(\theta)$, with $g(\theta) > 0$ if $\pi(\theta|y_{obs}) > 0$;
- an integer $N > 0$;
- a kernel function $K_h(u)$ and a scale parameter $h > 0$;
- a low dimensional vector of summary statistics $s = S(y)$.

ABC rejection sampling Algorithm

Sampling for $i = 1, ..., N$:

1. generate $\theta^{(i)} \sim g(\theta)$ from sampling density g;
2. generate $y \sim p(y|\theta^{(i)})$ from the likelihood;
3. compute summary statistic $s = S(y)$;
4. accept $\theta^{(i)}$ with probability $\frac{K_h(\|s - s_{obs}\|)\pi(\theta^{(i)})}{Kg(\theta^{(i)})}$, where $K \geq K_h(0) \max_\theta \frac{\pi(\theta)}{g(\theta)}$; else go to 1.

Output:

- a set of parameter vectors $\theta^{(1)}, ..., \theta^{(N)} \sim \pi_{ABC}(\theta|S_{obs})$.

Possibly, can add a stopping rule.

Conclusions

Next steps

Our focus till now was to understand the fundamental concepts and collect the missing information.

The next step will be a **simple and separate implementation** of both solution to be tested on simulated data.

Further steps will consider the **integration** of both solution into a single implementation and the testing on more complex data.

Bibliography

Pierre Jacob, John O'Leary, and Yves Atchadé.
Unbiased markov chain monte carlo with couplings.
Journal of the Royal Statistical Society: Series B (Statistical Methodology), 82, 08 2017.

Peter W. Glynn and Chang han Rhee.
Exact estimation for markov chain equilibrium expectations, 2014.

Jeffrey S. Rosenthal.
Faithful couplings of markov chains: Now equals forever.
Advances in Applied Mathematics, 18(3):372–381, 1997.

Dylan Cordaro.
Markov chain and coupling from the past.
2017.

Jinming Zhang.
Markov chains, mixing times and coupling methods with an application in social learning.
2020.

S. A. Sisson, Y. Fan, and M. A. Beaumont.
Overview of approximate bayesian computation, 2018.

Y. Fan and S. A. Sisson.
Abc samplers, 2018.

Dennis Prangle.
Summary statistics in approximate bayesian computation, 2015.