Improving Approximate Bayesian Computation with Pre-judgment Rule

Yanbo Wang, Xiaoqing Yu, Pinle Qin, Rui Chai, and Gangzhu Qiao(✉)

School of Data Science and Technology, North University of China,
Taiyuan 030051, Shanxi, China
qiaogangzhu@sohu.com

Abstract. Approximate Bayesian Computation (ABC) is a popular approach for Bayesian modeling, when these models exhibit an intractable likelihood. However, during each proposal of ABC, a great number of simulators are required and each simulation is always time-consuming. The overall goal of this work is to avoid inefficient computational cost of ABC. A pre-judgment rule (PJR) is proposed, which mainly aims to judge the acceptance condition using a small fraction of simulators instead of the whole simulators, thus achieving less computational complexity. In addition, it provided a theoretical study of the error bounded caused by PJR Strategy. Finally, the methodology was illustrated with various examples. The empirical results show both the effectiveness and efficiency of PJR compared with the previous methods.

Keywords: Approximate Bayesian Computation · Bayesian inference · Markov Chain Monte Carlo · Pre-judgment rule

1 Introduction

The crucial component of Bayesian statistics is to estimate the posterior distribution of parameter θ with given observations y. The posterior distribution, denoted as $p(\theta|y)$, satisfies that

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \propto p(y|\theta)p(\theta), \tag{1}$$

where $p(y) = \int p(y|\theta)p(\theta)d\theta$ is the normalizing constant and computationally inefficient in general. $p(y|\theta)$ and $p(\theta)$ represent likelihood function and the prior distribution, respectively. However, the likelihood $p(y|\theta)$ is not always intractable due to the lager sample size and high dimension of parameters. Approximate Bayesian Computation (ABC) methods provide likelihood-free approach for performing statistical inferences with Bayesian models [5,17,26]. The ABC method replaces the calculation of the likelihood function $p(y|\theta)$ in Eq. (1) with a simulation of the model that produces an artificial data set $\{x_i\}$. The most influential part of ABC is to construct some metric (or distance) and compare the simulated data $\{x_i\}$ to the observed data $\{y_i\}$ [6,15]. Recently, ABC has gained popularity particularly for the analysis of complex problems arising out of biological sciences (e.g. in population genetics, ecology, epidemiology, and systems biology) [5,24,27].

J. Zeng et al. (Eds.): ICPCSEE 2020, CCIS 1257, pp. 230–246, 2020.
https://doi.org/10.1007/978-981-15-7981-3_15

There are at least three leaps in the development of ABC, we denote as algorithms $\mathbb{A}$, $\mathbb{B}$ and $\mathbb{C}$. Algorithms of type $\mathbb{A}$, the simplest algorithm of ABC proposed in [25], is listed as follows:

- $\mathbb{A}1$. Sample θ from the prior distribution $p(\theta)$.
- $\mathbb{A}2$. Accept the proposed θ with probability h proportional to $p(y|\theta)$. Return to $\mathbb{A}1$.

Concretely, if θ^* denotes the maximum-likelihood estimator of θ, the acceptance probability h can be directly set as:

$$h = \frac{p(y|\theta)}{c}, \tag{2}$$

where c can be any constant greater than $p(y|\theta^*)$. Unfortunately, the likelihood function $p(y|\theta)$ is computationally expensive or even intractable. Hence Algorithm $\mathbb{A}1$ is not practical.

Many variants are proposed, among which one common approach is algorithms of type $\mathbb{B}$ [19]:

- $\mathbb{B}1$. Sample θ from the prior distribution $p(\theta)$.
- $\mathbb{B}2$. Generate x given the parameter θ via the simulator, i.e., $x \sim p(\cdot|\theta)$.
- $\mathbb{B}3$. Accept the proposed θ if $x = y$. Return to $\mathbb{B}1$.

The success of algorithm $\mathbb{B}$ depends on the fact that simulating from $p(\cdot|\theta)$ is easy for any θ, a basic assumption of ABC. To discriminate simulated data x from the observation y, we call x pseudo-observation here. Moreover, in Step $\mathbb{B}3$, $\mathbb{S}(x) = \mathbb{S}(y)$ is employed instead of $x = y$ in practice, where $\mathbb{S}(x)$ represents the summary statistics of x. It has been shown that if the statistics used in likelihood function are sufficient, then Algorithm $\mathbb{B}$ sample correctly from the true posterior distribution. Here, for ease of exposition, we use $x = y$ instead of $\mathbb{S}(x) = \mathbb{S}(y)$. Whereas the acceptance criteria $x = y$ is too restrictive here, leading the acceptance rate intolerably small. One might resort to relaxing the criteria as algorithm $\mathbb{C}$ [21]:

- $\mathbb{C}1$. Sample θ from the prior distribution $p(\theta)$.
- $\mathbb{C}2$. Generate x given the parameter θ via the simulator, i.e., $x \sim p(\cdot|\theta)$.
- $\mathbb{C}3$. Calculate the similarity between observations y and simulated data x, denoted $\rho(x, y)$[1].
- $\mathbb{C}4$. Accept the proposed θ if $\rho(x, y) \geq \xi$ (ξ is a prespecified threshold). Return to $\mathbb{C}1$.

Notice that in Step $\mathbb{C}2$, a quantity of pseudo-observations x are simulated from $p(\cdot|\theta)$ independently, i.e., $x = \{x_1, ..., x_S\}, x_i \sim p(\cdot|\theta)$ $i.i.d.$, where S is the number of simulators in each proposal and always fixed, independent of θ. The similarity $\rho(x, y)$ can be represented in terms of the average similarity between x_i and y such that $\rho(x, y) = \frac{1}{S}\sum_{i=1}^{S} \pi_\zeta(x_i|y)$, where $\pi_\zeta(\cdot|y)$ is an ζ-kernel around observation y[2].

It is apparent that the choice of S plays a critical role in the efficiency of the algorithm. Obviously a large S will degrade the efficiency of ABC. In contrast, if S is small,

[1] $\rho(\mathbb{S}(x), \mathbb{S}(y))$ is replaced by $\rho(x, y)$, similar with Step $\mathbb{C}3$.

[2] E.g., ζ-kernel can be chosen as $\pi_\zeta(x_1|x_2) = (1/\sqrt{2\pi}\zeta)\exp(-\|x_1 - x_2\|^2/2\zeta^2)$.

though leading a significant reduction for each θ in computation, the samples may fail to converge to the target distribution [4]. Moreover, it is awful to spend amounts of computation (S simulations) for just 1 bit information, namely accept or reject the proposal. A natural question is proposed: can we simulate a small number of pseudo-observations in Step $\mathbb{C}2$ and maintain the convergence to the target distribution simultaneously? Or can we find a tradeoff between efficiency and accuracy? Here, we claim it is feasible.

In this paper, we devise Pre-judgment (PJR) rule, adjusting number of simulators dynamically, instead of using a constant S. In short, we firstly generate small amount of data and estimate a rough similarity. If the similarity is far away from the prespecified threshold (say, in Step $\mathbb{C}4$, ξ), then we judge (accept/reject) the proposal ahead. Otherwise, we draw more data from the simulator and repeat the evaluation until we have enough evidence to make the decision. Empirical results show that majority of these decision can be made based on a small amount of simulators with high confidence, thus lots of computations are saved.

The remainder of the paper is organized as follows. Section 2 describes our algorithm and Sect. 3 provides theoretical analysis. A toy model is shown in Sect. 4.1 to show some properties of PJR based method. Furthermore, the empirical evaluations are given in Sect. 4.2. Finally, the last section is devoted to conclude the paper.

2 Methodology

In this section, we will review the relative works and then present our method. Firstly, we introduce how pre-judgment rule (PJR) accelerate ABC rejection method. Then we adapt PJR strategy to ABC-MCMC framework [20].

2.1 Related Works

In this section, we briefly review the related studies. Firstly, we focus on recent developments in ABC community. Though allowing parallel computation, ABC is still in its infancy owing to the large computational cost. Many approaches are proposed to scale up ABC in machine learning community. Concretely, [22,29] introduced Gaussian process to accelerate ABC. [23] made use of the random seed in sampling procedure and transform ABC sampler into an deterministic optimization procedure. [21] adapted Hamiltonian Monte Carlo to ABC scenario, allowing noise in estimated gradient of log-likelihood by borrowing the idea from stochastic gradient MCMC framework [1,2,11,12,18,28] and pseudo-marginal MCMC methods [3,14].

In addition, theoretical works has become popular recently [4,7,8,30]. Some works focus on the selection of summary statistics [9,13]. Different from these methods, PJR strategy essentially alleviates the computational burden in ABC rejection step, which can be extended to any ABC scenario, e.g., ABC rejection approach and ABC-MCMC proposed in this paper.

2.2 PJR Based ABC: (PJR-ABC)

In the Algorithm A, the likelihood is not available explicitly. Thus we resort to approximate methods by introducing the simulated data x, as follows:

$$p(y|\theta) = \int \delta_D(x-y)p(x|\theta)dx \approx \int \pi_\zeta(x|y)p(x|\theta)dx \approx \frac{1}{S}\sum_{i=1}^{S}\pi_\zeta(x_i|y), \tag{3}$$

where $\delta_D(\cdot)$ is the Dirac delta function. Then a relaxation is employed by introducing an ζ-kernel around the observation y. The last approximate equality use a Monte Carlo estimate of the likelihood via S draws of x from simulator $p(\cdot|\theta)$.

On the other hand, for Algorithm $\mathbb{C}$, the similarity between pseudo-observations x and raw observations y can be expressed as the mean similarity between each simulator output x_i and y

$$\rho(x,y) = \frac{1}{S}\sum_{i=1}^{S}\pi_\zeta(x_i|y). \tag{4}$$

From Eq. (3) and (4), it is validated that Algorithm $\mathbb{A}$ is equivalent to Algorithm $\mathbb{C}$ in essence. Then acceptance conditions in both Step $\mathbb{A}2$ and Step $\mathbb{C}4$ are equivalent to performing a comparison (between z and z_0, defined later). Specifically, firstly we compute $z = \frac{1}{S}\sum_{i=1}^{S}\pi_\zeta(x_i|y)$, where $x_i \sim p(\cdot|\theta)$ $i.i.d.$, and then compare it with z_0, a constant. If $z > z_0$, accept the proposed θ. If $z \leq z_0$, reject it, where z_0 is a prespecified threshold, say, in Step $\mathbb{C}4$, z_0 corresponds to ξ[3].

To guarantee the convergence to the true posterior, S should be a large number, which means each proposal needs S simulations [4]. However, spending quantities of computation (i.e., simulating S pseudo-data $x_1, \ldots, x_S$) to get just one bit of information, namely whether to accept or reject a proposal, is likely not the best use of computational resources.

To address this issue, PJR is devised to speedup the ABC procedure. We are willing to tolerate small error in this step to achieve faster judgement. In particular, we firstly draw a small number of pseudo-observations x and estimate a rough z. If the difference between z and z_0 is significantly larger than the standard deviation of z, we claim that z is far away enough from z_0 confidently and make the decision by comparing the rough z with z_0. Otherwise, we draw more pseudo-observations to increase the precision of z until we have enough evidence to make the decision.

More formally, checking the acceptance condition can be reformulated to the following statistical hypothesis test.

$$H_1 : z > z_0, \quad H_2 : z \leq z_0.$$

In order to test the hypothesis, we are able to generate infinitely many pseudo-observations from $p(\cdot|\theta)$. On the other hand, we expect to simulate less pseudo-observations owing to computational cost.

[3] In Step A2, z_0 is more complex. Checking the acceptance condition is equivalent to judging $\frac{z}{c} > u$, where c is defined in Eq. 2 and $u \sim \text{Uniform}(0,1)$.

To do this, we proceed as follows. We compute the sample mean $\bar{z}$ and sample standard deviation s_z as

$$z_i = \pi_\zeta(x_i|y), \quad \bar{z} = \frac{1}{n} z_i, \quad s_z = \sqrt{\frac{\bar{z^2} - (\bar{z})^2}{n-1}}, \tag{5}$$

where $\bar{z^2}$ represents the mean of z^2. Then we compute the test statistics t via

$$t = \frac{\bar{z} - z_0}{s_z}. \tag{6}$$

It is assumed that n is large enough here. Under this situation central limit theorem (CLT) kicks in and the test statistic t follows the standard Student-t distribution with $n-1$ degrees of freedom. Note that when n is large enough, Student-t distribution with $n-1$ degrees of freedom is close to the standard normal distribution. Then we compute η defined as:

$$\eta = 1 - \psi_{n-1}(|t|), \tag{7}$$

where $\psi_{n-1}(\cdot)$ is the cdf of the standard Student-t distribution with $n-1$ degrees of freedom.

Then we provide a threshold ϵ, e.g., $\epsilon = 0.1$. If $\eta < \epsilon$, we make a decision that z is significantly different from z_0. Then we accept/reject θ via comparing $\bar{z}$ and z_0. If $\eta \geq \epsilon$, it means that we do not have enough evidence to decide. Thus more pseudo-observations are drawn to reduce the uncertainty of z. Note that when S pseudo-observations are drawn, the procedure would be terminated and it reduces to previous ABC algorithm. The resulting algorithm can be seen in Algorithm 1.

The advantage of PJR-ABC is that we can often make confident decisions with s_i ($s_i \ll S$) pseudo-observations and reduce computation significantly. Though PJR-ABC brings error in judgement, we can use the computational time we save to draw more samples to offset the small bias. Worth to note that ϵ can be regarded as a knob. When ϵ approaches to 0, we make almost the same decision with the ABC rejection method but requires masses of simulators. On the other hand, when ϵ is high, we make decisions without sufficient evidence and the error would be high. This accuracy-efficiency trade-off will be empirically verified in Sect. 4.1.

2.3 PJR Based Markov Chain Monte Carlo Version of ABC: PJR-ABC-MCMC

The ABC rejection methods are easy to implement and compatible with embarrassingly parallel computation. However, when the prior distribution is long way from posterior distribution, most of the samples from prior distribution would be rejected, leading acceptance rate too small, especially in high-dimensional problem. To address this issue, a Markov Chain Monte Carlo version of ABC (ABC-MCMC) algorithm is proposed [20]. It is well-known that MCMC has been the main workhorse of Bayesian computation since 1990s and many state-of-the-art samplers in MCMC framework can be extended into ABC scenario, e.g., Hamiltonian Monte Carlo can be extended to Hamiltonian ABC [21]. Hence ABC-MCMC [20] is a benchmark in ABC community. Now we show that our PJR rule can be adapted to the ABC-MCMC framework. First, ABC-MCMC is briefly introduced:

Algorithm 1. PJR-ABC

Require: θ drawn from prior $p(\theta)$, $\{s_i\}_{i=0}^{k}$: a strictly increasing sequence satisfying that $s_i \in \mathbb{N}_+$, $s_0 = 0$ and $s_k = S$. knob ϵ.
Ensure: accept/reject θ

for $i = 1 : k$ **do**
  draw $|s_i - s_{i-1}|$ pseudo-observations $x_{s_{i-1}+1}, x_{s_{i-1}+2}, \ldots, x_{s_i}$ from simulator $p(\cdot|\theta)$, compute the corresponding $z_{s_{i-1}+1}, \ldots, z_{s_i}$ and store, where $z_i = \pi_\zeta(x_i|y)$.
  Set $n = s_i$.
  Update the mean $\bar{z}$ and std s_z using Equation (5).
  Compute the test statistics t via Equation (6).
  Compute η via Equation (7).
  if $\eta < \epsilon$ **then**
    if $\bar{z} > z_0$ **then**
      accept the proposed θ and break.
    else
      reject the proposed θ and break.
    end if
  end if
end for
if $\bar{z} > z_0$ **then**
  accept the proposed θ.
else
  reject the proposed θ.
end if

- $\mathbb{D}1$. Given the current point θ, θ' is proposed according to a transition kernel $q(\theta'|\theta)$.
- $\mathbb{D}2$. Generate x' from the simulator $p(\cdot|\theta')$.
- $\mathbb{D}3$. Compute the acceptance probability α defined in Eq. 8.
- $\mathbb{D}4$. Accept θ' with probability α. Otherwise, stay at θ. Return to $\mathbb{D}1$.

In MCMC sampler, MH acceptance probability α is defined as

$$\alpha = \min\left\{1, \frac{p(\theta')p(y|\theta')q(\theta|\theta')}{p(\theta)p(y|\theta)q(\theta'|\theta)}\right\}. \tag{8}$$

In likelihood-free scenario, the acceptance probability of ABC-MCMC is

$$\alpha = \min\left\{1, \frac{p(\theta')\sum_{s=1}^{S}\pi_\zeta(x'_s|y)q(\theta|\theta')}{p(\theta)\sum_{s=1}^{S}\pi_\zeta(x_s|y)q(\theta'|\theta)}\right\},$$

where $x_s \sim p(\cdot|\theta)$ $i.i.d.$ and $x'_s \sim p(\cdot|\theta')$ $i.i.d.$ The acceptance of proposal is determined by following form:

$$u < \alpha = \min\left\{1, \frac{p(\theta')\sum_{s=1}^{S}\pi_\zeta(x'_s|y)q(\theta|\theta')}{p(\theta)\sum_{s=1}^{S}\pi_\zeta(x_s|y)q(\theta'|\theta)}\right\},$$

where $u \sim \text{Uniform}(0,1)$. This is equivalent to the following expression:

$$u < \frac{p(\theta')\frac{1}{S}\sum_{s=1}^{S}\pi_\zeta(x'_s|y)q(\theta|\theta')}{p(\theta)\frac{1}{S}\sum_{s=1}^{S}\pi_\zeta(x_s|y)q(\theta'|\theta)}.$$

Note that $\{x_1, ..., x_S\}$ is given in ABC-MCMC, then define the fixed part z_0 and test variable z, we obtain that

$$z_0 = \frac{p(\theta)}{p(\theta')}\frac{1}{S}\sum_{s=1}^{S}\pi_\zeta(x_s|y)\frac{q(\theta'|\theta)}{q(\theta|\theta')}u, \;\; z = \frac{1}{S}\sum_{s=1}^{S}\pi_\zeta(x'_s|y),$$

where z can be further simplified into the following form, similar to PJR-ABC: $z = \frac{1}{S}\sum_{i=1}^{S} z_i$, where $z_i = \pi_\zeta(x'_i|y)$.

Following PJR-ABC, we test the following hypothesis $H_1 : z_0 > z$ vs $H_2 : z_0 < z$. Then the sample mean $\bar{z}$, the sample standard deviation s_z and the test statistics t can be calculated as shown in Eq. (5) and (6), same with PJR-ABC. The resulting algorithm is similar and not listed.

3 Theoretical Analysis

In this section, we study the theoretical properties for PJR strategy. Specifically, we provide the error analysis for both PJR-ABC and PJR-ABC-MCMC. Since every time we accept/reject a proposal in PJR-ABC/PJR-ABC-MCMC, we deal with a hypothesis testing problem. We are attempting to bound the error caused by such a testing problem first. Then we build the relationship between such a single test error and total error for both PJR-ABC and PJR-ABC-MCMC. Now we focus on the error caused by a single testing problem. In hypothesis testing problem, two types of error are distinguished. A type I error is the incorrect rejection of a true hypothesis while the type II error is the failure to reject a false hypothesis. Now we discuss the probabilities of these two errors in a single decision problem.

Theorem 1. *The probability of both the error I and II decreases approximately exponentially w.r.t. the sample size of z (sample size of z corresponds to $s_1, \ldots, s_k$ in Algorithm 1).*

Proof. We assume that $\psi_{n-1}(\cdot)$ is the cdf of standard Student-t distribution with degree $n-1$. For simplicity, we first discuss the probability of type I error, i.e., the incorrect rejection of a true hypothesis. It would be easy to extend the conclusion into the type II error owing to the symmetry.

In this case, $z > z_0$. Suppose the number of sampled z is n. The test statistics t satisfies that $t = \frac{\bar{z}-z_0}{s_z}$, following the standard Student-t distribution with degree $n-1$. The standard Student-t distribution is approaching to the standard normal distribution when the degree $n-1$ is large enough. Hence, many properties of normal distribution can be shared.

Given the knob parameter ϵ, according to the monotonicity of the function $\psi_{n-1}(\cdot)$ on $\mathbb{R}$, we know that there exists a unique s such that $\psi_{n-1}(s) = \epsilon$. Moreover, since $\bar{z} = \frac{z_1+z_2+\ldots+z_n}{n}$ and $t = \frac{\bar{z}-z_0}{s_z} \sim \psi_{n-1}(\cdot) \approx \mathcal{N}(0,1)$, we have that z_i can be seen as sampled independent identically distributed from $\mathcal{N}(z_0, ns_z)$, i.e., $z_i \sim \mathcal{N}(z_0, ns_z)$ $i.i.d.$

The type I error only occurs when $\frac{\bar{z}-z_0}{s_z} < s$. That is, $\sum_{i=1}^{n} z_i < n(s_z s + z_0)$. Thus, we can have the probability of type I error via integrating over the space $(z_1, z_2, \ldots, z_n)$ and $\sum_{i=1}^{n} z_i < n(s_z s + z_0)$.

$$
\begin{aligned}
&\Pr\ (\text{Type I error}) = \Pr(\textstyle\sum_{i=1}^{n} z_i < n(s_z s + z_0)) \\
&= \int \cdots \int \int_{-\infty}^{(z_1,z_2,\cdots,z_n),\sum_{i=1}^{n} z_i < n(s_z s+z_0)} \psi'(z_1)\psi'(z_2)\ldots\psi'(z_n)dz_1 dz_2 \ldots dz_n \\
&= \int_{-\infty}^{n(s_z s+z_0)-z_1-\ldots-z_{n-1}} \cdots \int \int_{-\infty}^{z_1} \psi'(z_1)\psi'(z_2)\ldots\psi'(z_n)dz_1 dz_2 \cdots dz_n \\
&= \psi_{n-1}(z_1)\psi_{n-1}(z_2)\ldots\psi_{n-1}(n(s_z s + z_0) - z_1 - \ldots - z_{n-1})
\end{aligned}
$$

where $\psi'(\cdot)$ and $\psi_{n-1}(\cdot)$ represent the pdf and cdf of the standard Student-t distribution with $n-1$ degree of freedom.

This completes the proof.

The above theorem demonstrates that the error during a single judge can be negligible as long as the number of sampled z is large enough. Based on this theorem, the following assumption are reasonable.

Assumption 1. The probability of error produced by a single hypothesis testing problem in both PJR-ABC and PJR-ABC-MCMC can be upper-bounded, denoted by $\delta_1, \delta_2 \to 0_+$, for PJR-ABC and PJR-ABC-MCMC, respectively.

In Bayesian inference, we are interested in the posterior average, defined as $\bar{\phi} \triangleq \int_\theta \phi(\theta)p(\theta|y)d\theta$ for some test function $\phi(\theta)$ of interest. For a given numerical method (say, PJE-ABC or PJR-ABC-MCMC) with generated samples $\{\theta_1, \ldots, \theta_M\}$, we use the sample average $\hat{\phi}$ defined as $\hat{\phi} = 1/M \sum_{l=1}^{M} \phi(\theta_l)$ to approximate $\bar{\phi}$. Before providing a bound for the bias of a PJR-ABC algorithm, we make a mild assumption first.

Assumption 2. The prior average of $\phi(\cdot)$ is bounded away from infinity, i.e.,

$$\int_\theta \phi(\theta)p(\theta)d\theta < +\infty.$$

Theorem 2. *Under Assumption (1) and (2), the bias of PJR-ABC can be upper-bounded as:* $|\mathbb{E}\hat{\phi} - \bar{\phi}| \leq C_1\delta_1$, *where* $C_1 = \frac{\int_\theta \phi(\theta)p(\theta)d\theta}{p(y)}$ *is a constant,* $p(y)$ *denotes the normalizing constant.*

Proof. In ABC rejection method, each θ drawn from $p(\theta)$ is independent. The error at θ caused by PJR is denoted by $\xi(\theta)$, which is assumed to be a perturbation on the true likelihood. Thus the estimated likelihood function can be represented as $\hat{p}(y|\theta) = p(y|\theta) + \xi(\theta)$, where $|\xi(\theta)| \leq \delta_1$ owing to the boundedness of single error, described in Assumption 3.

$$
\begin{aligned}
\mathbb{E}(\hat{\phi}) &= \tfrac{1}{p(y)} \textstyle\int \phi(\theta)\hat{p}(y|\theta)p(\theta)d\theta \\
&= \tfrac{1}{p(y)} \textstyle\int \phi(\theta)p(y|\theta)p(\theta)d\theta + \tfrac{1}{p(y)} \int \phi(\theta)\xi(\theta)p(\theta)d\theta
\end{aligned} \tag{9}
$$

The first term in RHS of Eq. (9) is the expectation of the true posterior distribution. While the second term is the error. We can observe that the error is upper bounded.

$$\frac{1}{p(y)} \int \phi(\theta)\xi(\theta)p(\theta)d\theta \leq \frac{1}{p(y)}|\delta_1| \int \phi(\theta)p(\theta)d\theta = C_1|\delta_1|,$$

where $C_1 = \frac{\int \phi(\theta)p(\theta)d\theta}{p(y)}$ is bounded followed from the fact that both $\frac{1}{p(y)}$ and $\int \phi(\theta)p(\theta)d\theta$ are bounded away from $+\infty$.

This completes the proof.

In PJR-ABC, each sample is independent with each other. However, in PJR-ABC-MCMC, all the samples are in a single chain, leading the analysis more complicated. Here, the distance between probability distributions is measured by the total variational distance (TVD),[4] described as follows.

Theorem 3. *Under Assumption 3, for any posterior distribution, there exists a constant C_2 such that the discrepancies between the true posterior distribution $\mathcal{S}_0$ and the stationary distribution of our PJR-ABC-MCMC algorithm $\mathcal{S}_\epsilon$ can be upper bounded as: $d_v(S_0, S_\epsilon) \leq C_2\delta_2$.*

Proof. We firstly focus on the error for a single step. Based on this, the error about the stationary distribution is derived. The transition kernel of the ABC-MCMC algorithm can be written as

$$\mathcal{T}_0(\theta, \theta') = P_a(\theta, \theta')q(\theta'|\theta) + (1 - P_a(\theta, \theta'))\delta_D(\theta' - \theta),$$

where $\delta_D(\cdot)$ is the Dirac delta function, $P_a(\theta, \theta')$ is the acceptance probability. Similar definition of transition kernel of PJR-ABC-MCMC hold for $\mathcal{T}_\epsilon(\theta, \theta')$ and acceptance probability $P_{a,\epsilon}(\theta, \theta')$.

The discrepancies between $P_a(\theta, \theta')$ and $P_{a,\epsilon}(\theta, \theta')$ is defined as: $\delta P_a(\theta, \theta') \triangleq P_{a,\epsilon}(\theta, \theta') - P_a(\theta, \theta')$. For every (θ, θ'), according to the error for a single test, there exists an upper bound for $\delta P_a(\theta, \theta')$, i.e., $|\delta P(\theta, \theta')| \leq \delta_{\max}$ for $\forall\ (\theta, \theta')$.

Then the total variational distance for a single step can be upper bounded for any distribution P as:

$$\begin{aligned}
\int_{\theta'} |(\mathcal{P}\mathcal{T}_\epsilon)(\theta') - (\mathcal{P}\mathcal{T}_0)(\theta')|d\Omega(\theta') &= \int_{\theta'} | \int_\theta (\mathcal{T}_0(\theta, \theta') - \mathcal{T}_\epsilon(\theta, \theta'))dP(\theta)|d\Omega(\theta') \\
&= \int_{\theta'} | \int_\theta (\mathcal{T}_0(\theta, \theta') - \mathcal{T}_\epsilon(\theta, \theta'))dP(\theta)|d\Omega(\theta') \\
&= \int_{\theta'} | \int_\theta (q(\theta'|\theta) - \delta_D(\theta' - \theta))(\delta P(\theta, \theta'))dP(\theta)|d\Omega(\theta') \\
&\leq \int_{\theta'} \int_\theta |(q(\theta'|\theta) - \delta_D(\theta' - \theta)| \cdot |\delta_{\max}| \cdot dP(\theta)d\Omega(\theta') \\
&\leq \delta_{\max} \int_{\theta'} | \int_\theta q(\theta'|\theta)dP(\theta')|d\Omega(\theta') \\
&+ \delta_{\max} \int_{\theta'} | \int_\theta \delta_D(\theta' - \theta)dP(\theta')|d\Omega(\theta') = 2\delta_{\max}
\end{aligned}$$

Then apply Lemma 1, substitute $2\delta_{\max}$ into δ in Eq. 10 we prove Theorem 3. This completes the proof.

[4] The total variation distance between two distribution P and Q, absolutely continuous w.r.t. measure Ω, is defined as $d_v(P, Q) \triangleq 1/2 \int_\theta |f_P(\theta) - f_Q(\theta)|d\Omega(\theta)$, where $f_P(\cdot)$ and $f_Q(\cdot)$ are their respective densities.

Lemma 1 [16]. *Given two transition kernels, $\mathcal{T}_0$ and $\mathcal{T}_\epsilon$, whose stationary distributions are denoted by $\mathcal{S}_0$ and $\mathcal{S}_\epsilon$, if $\mathcal{T}_0$ satisfies the following contraction condition with a constant $\eta \in [0, 1)$ for all probability distribution $\mathcal{P}$:*

$$d_v(\mathcal{P}\mathcal{T}_0, \mathcal{S}_0) \leq \eta d_v(\mathcal{P}, \mathcal{S}_0)$$

and the one step error between $\mathcal{T}_0$ and $\mathcal{T}_\epsilon$ is upper bounded uniformly with a constant $\delta > 0$ as:

$$d_v(\mathcal{P}\mathcal{T}_0, \mathcal{P}\mathcal{T}_\epsilon) \leq \delta, \forall \mathcal{P} \tag{10}$$

then the distance between $\mathcal{S}_0$ and $\mathcal{S}_\epsilon$ can be bounded as: $d_v(S_0, S_\epsilon) \leq \frac{\delta}{1-\eta}$

Theorem 2 and 3 indicate that the error is proportional to the single testing error. Combining this result with Theorem 1, we know that the bias of both PJR-ABC and PJR-ABC-MCMC can be bounded.

4 Numerical Validation

In this section, we use a toy model to demonstrate both PJR-ABC and PJR-ABC-MCMC.

4.1 Synthetic Data

We adopt the gamma prior with shape α and rate β, i.e., $p(\theta) = \text{Gamma}(\alpha, \beta)$. The likelihood function is exponential distribution, i.e., $x \sim \exp(1/\theta)$. Let observations are generated via $y = \frac{1}{N}\sum_{i=1}^{N} e_i$, where $e_i \sim \exp(1/\theta^*)$, N is the number of observations. Regarding the selection of the sequence $\{s_i\}_{i=1}^{k}$ ($s_0 = 0$), we find geometric sequence is the usually the best choice, thus is used in both Sect. 4.1 and 4.2. The common ratio of the geometric sequence is usually set to 1.5–2. The true posterior is a gamma distribution with shape $\alpha + N$ and rate $\beta + Ny$, i.e., $p(\theta|y) = \text{Gamma}(\alpha + N, \beta + Ny)$. In particular, we set $S = 1000$, $N = 20$, $y = 7.74$, $\alpha = \beta = 1$, $\theta^* = 0.15$ in this scenario. We run chains of length 50K for ABC-MCMC and PJR-ABC-MCMC and 100K for ABC and PJR-ABC. For each method, we conduct 5 independent trials and report the average value. In this paper, the choice of proposal distribution in both ABC-MCMC and PJR-ABC-MCMC is a Gaussian distribution centered at current θ.

First, we investigate how the performance (both efficiency and accuracy) changes as a function of the knob ϵ empirically. For each $\epsilon \in \{0, 0.01, 0.03, 0.07, 0.1, 0.2, 0.3\}$, we record both efficiency[5] and accuracy[6]. $\epsilon = 0$ means the PJR-ABC/PJR-ABC-MCMC reduce to ABC/ABC-MCMC approach. The results are reported in Fig. 1. We find that smaller ϵ usually leads to higher accuracy and less efficiency, validating the statement about ϵ mentioned in Sect. 2. Hence, the empirical trade-off between efficiency and accuracy can be controlled by adjusting ϵ. In the following, we set $\epsilon = 0.1$. In Fig. 3, we show the trace plots of the last 1K samples from a single chain for ABC-MCMC

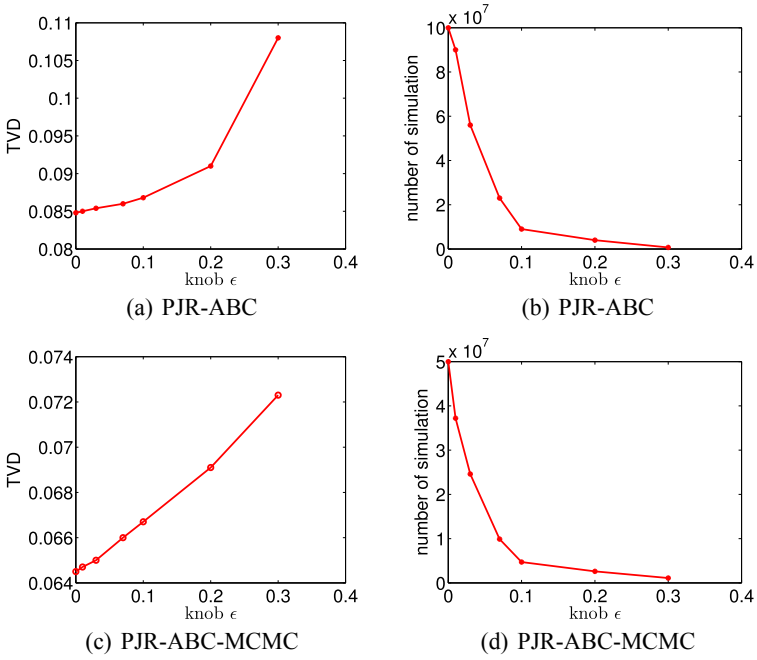
[5] Measured in term of number of simulator.

[6] Measured in term of TVD with the true posterior distribution.

and PJR-ABC-MCMC. It is a positive result, indicating PJR-ABC-MCMC preserve the ability of exploration to the parameter space compared with ABC-MCMC. The empirical histograms of θ for all the methods are presented in Fig. 2. We find that all of them are close to the desired posterior. In Table 1 we show

- the average Total Variational Distance[7] (between the true posterior and the ABC posteriors) and the corresponding standard deviation using the first 10K samples and whole chain;
- the average number of simulators.

We can observe that our PJR based ABC rejection and ABC-MCMC achieve similar result with original algorithm in convergence to the target posterior distribution. Furthermore, PJR strategy can accelerate both ABC and ABC-MCMC in terms of number of simulators.



(a) PJR-ABC (b) PJR-ABC

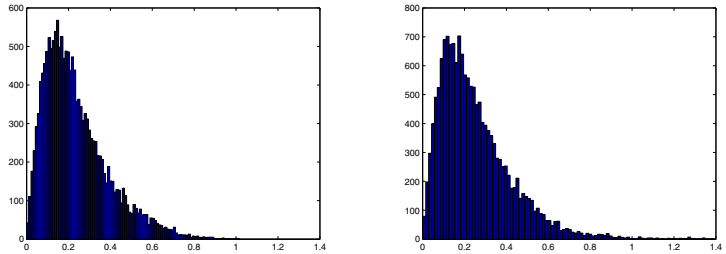
(c) PJR-ABC-MCMC (d) PJR-ABC-MCMC

Fig. 1. Demonstration problem. TVD and number of simulations as a function of the knob ϵ.

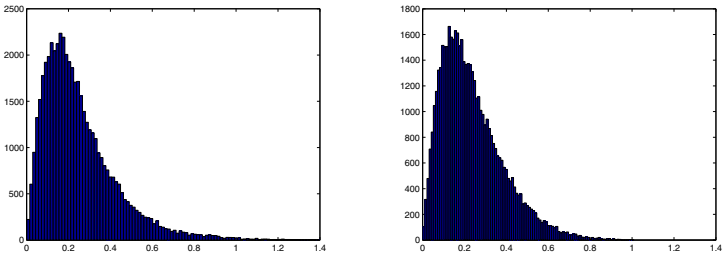
4.2 Real Applications

The Popular Ricker Model. In this section, we show the application of our method on the popular Ricker model [31]. The Ricker model, a classic discrete population model

[7] Note that in experiment the total variational distance is estimated empirically owing to the absence of explicit formulae.

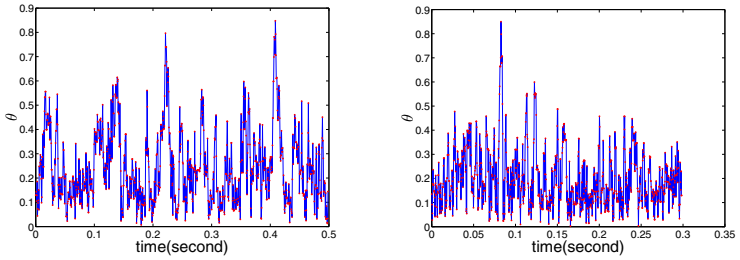


(a) ABC (b) PJR-ABC



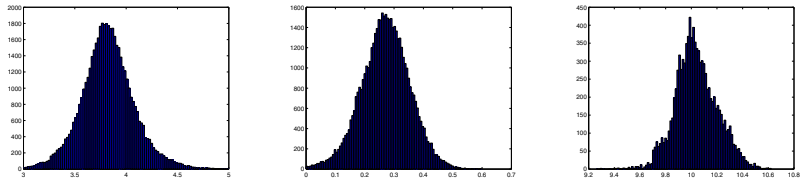
(c) ABC-MCMC (d) PJR-ABC-MCMC

Fig. 2. Demonstration problem. The empirical histograms of θ for all the methods.



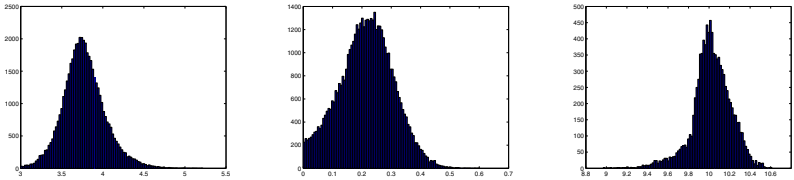
(a) ABC-MCMC (b) PJR-ABC-MCMC

Fig. 3. Demonstration problem. Trace plot of last 1K samples, where $\epsilon = 0.1$.



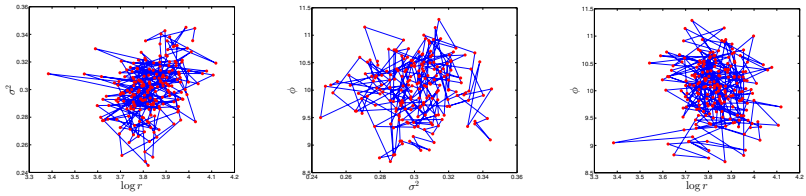
(a) $\log r$ (b) σ^2 (c) ϕ

Fig. 4. Ricker model. Empirical histogram of parameter $\theta = (\log r, \sigma, \phi)$ generated by ABC-MCMC.



(a) $\log r$ (b) σ^2 (c) ϕ

Fig. 5. Ricker model. Empirical histogram of parameter $\theta = (\log r, \sigma, \phi)$ generated by PJR-ABC-MCMC.



(a) $\log r - \sigma^2$ (b) $\sigma^2 - \phi$ (c) $\log r - \phi$

Fig. 6. Ricker model. Trajectories of each pair of two parameters over the last 200 time-steps generated by our PJR-ABC-MCMC.

Table 1. Results for the demonstration problem in terms of TVD (Total Variational Distance) and number of simulators. Note that for TVD the value below is the actual value times 100 (mean $\pm$ std). Simulators represent the total number of pseudo-observations from the simulator. For the first two approaches, we draw 100K samples while for the last two approaches, 50K samples are drawn.

Algorithm	10K	Whole chain	Simulators
ABC	8.48 ± 0.89	6.15 ± 0.03	100M
PJR-ABC	8.68 ± 0.45	6.11 ± 0.02	**9M**
ABC-MCMC	6.43 ± 0.03	5.96 ± 0.04	50M
PJR-ABC-MCMC	6.48 ± 0.07	6.03 ± 0.04	**4.7M**

used in ecology, gives the expected number of individuals in current generation as a function of number of individuals in previous generation. This model is commonly used as an exampler of complex model [29] because it cause the collapse of standard statistical methods due to near-chaotic dynamics [31]. In particular, N_t denote the unobserved number of individuals in the population at time t while the number of observed individuals is denoted by Y_t. The Ricker model is defined via the following relationships [31]

$$\begin{aligned} N_{t+1} &= rN_t\exp(-N_t + e_t), \;\; Y_t \sim \text{Poisson}(\phi N_t), \\ e_t &\sim \quad \mathcal{N}(0, \sigma^2), \end{aligned}$$

where each e_t $(t = 1, 2, ...,)$ is independent and Y_t only depends on N_t. In this model, the parameter vector is $\theta = \{\log r, \sigma^2, \phi\}$. $y_{1:T} = \{y_1, ..., y_T\} \in \mathbb{R}^T$ is the time-series of observations. For each parameter, we adopt the uniform prior as

$$\begin{aligned} \log r &\sim \text{Uniform}(3, 6), \quad \sigma \sim \text{Uniform}(0, 0.8), \\ \phi &\sim \quad \text{Uniform}(4, 20). \end{aligned}$$

The target distribution is the posterior of θ given observations $y_{1:T}$, i.e., $p(\theta|y_{1:T})$. Artificial dataset is generated using $\theta^* = (3.8, 0.3, 10.0)$. We compare PJR-ABC-MCMC method with ABC-MCMC. For ABC-MCMC, we run the simulator $S = 2000$ times at each θ to approximate the likelihood value. The knob ϵ is set to be 0.1. For summary statistics, we follow the methods described in [29], which contain a collection of phase-invariant measures, such as coefficients of polynomial autogressive models.

Effectiveness: Figure 4 and 5 show the empirical histogram of parameter of interest $\theta = (\log r, \sigma, \phi)$ generated by ABC-MCMC and PJR-ABC-MCMC, respectively. Furthermore, we present the scatter plots of trajectories for every two parameters in Fig. 6. We can observe that the mode of the empirical posterior is close to the θ^* and the posteriors produced by the two algorithms are similar, showing the success of PJR-ABC-MCMC in Ricker model.

Efficiency: The simulation procedure is complex and dominate in computational time. Therefore, the running time of samplers is almost proportional to the number of simulators. Specifically, sampling 1K parameters, ABC-MCMC requires 2M simulators $(S = 2000)$ while PJR-ABC-MCMC only requires about 371K simulators. We conclude that majority of the decision can be made based on a small amount of simulators with high confidence. Hence, our PJR strategy accelerates ABC-MCMC algorithm greatly in Ricker model.

4.3 Apply to HABC-SGLD

In this part, we apply our method to SGLD (Stochastic Gradient Langevin Dynamics, [28]) version of HABC (Hamiltonian ABC) proposed in [21].

In each iteration of SGLD, a mini-batch $\mathcal{X}_n$ of size n is drawn to estimate the gradient of log-posterior. The proposal is

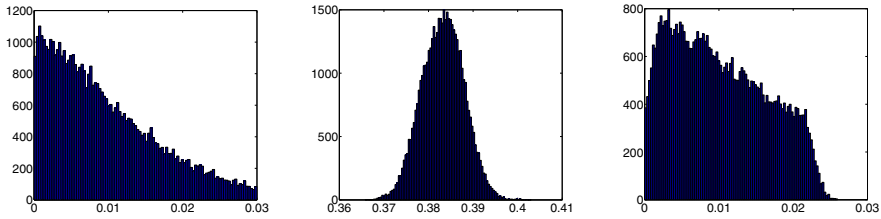
$$\theta' \sim q(\cdot|\theta, \mathcal{X}_n) = \mathcal{N}(\theta + \tfrac{\alpha}{2}\nabla_\theta\{\tfrac{N}{n}\textstyle\sum_{i\in\mathcal{X}_n} \log p(x_i|\theta) + \log p(\theta)\}, \alpha)$$

It can be shown that when the stepsize α approaches to zero, the acceptance probability approaches to 1 [28]. Based on this, the MH correction step is ignored. However, the assumption that $\alpha \to 0$ is too restrictive. In practice, to keep the mixing rate high, we always choose a reasonably large α. Under this situation, SGLD can not converge to target distribution in some cases. The detailed reasons can be found in [16].

In ABC scenarios, conventional MH rejection step is time-consuming. So our method fit to this problem naturally. Specifically, we consider an L1-regularized linear regression model. This model has been used in [16] to explain the necessity of MH rejection in SGLD. We explore its effectiveness in ABC scenario.

Given a dataset $\{u_i, v_i\}_{i=1}^N$, where u_i are the predictors and v_i are the targets. Gaussian error model and Laplacian prior for parameter $\theta \in \mathbb{R}^D$ are adopted, i.e., $p(v|u,\theta) \propto \exp(-\frac{\lambda}{2}(v-\theta^T u)^2)$ and $p(\theta) \propto \exp(-\lambda_0\|\theta\|_1)$. We generate a synthetic dataset of size $N = 10000$ via $v_i = \theta_0^T u_i + \xi$, where $\xi \sim \mathcal{N}(0, 1/3)$ and $\theta_0 = 0.5$, following [16]. For pedagogical reason, we set $D = 1$. Furthermore, we choose $\lambda = 1$ and $\lambda_0 = 4700$ so that the prior is not washed out by the likelihood.

Here, standard MCMC sampler is employed as the baseline method. And we run the HABC-SGLD without rejection and HABC-SGLD with rejection (PJR-HABC-SGLD). The empirical histograms of samples obtained by running different samplers are shown in Fig. 7. We observe that the empirical histogram of samples obtained from PJR-HABC-SGLD is much closer to the standard MCMC sampler than that of HABC-SGLD, thus verifying the effectiveness of PJR-HABC-SGLD.



(a) standard MCMC sampler (b) HABC-SGLD (c) PJR-HABC-SGLD

Fig. 7. Application to HABC-SGLD. Empirical histogram of samples obtained by different samplers. We can observe that HABC-SGLD fails to converge to the posterior distribution in this situation. But PJR correction version of HABC-SGLD converges to the posterior.

5 Conclusion

In this paper, we have proposed pre-judgment Rule to accelerate ABC method. Computational methods adaptive to ABC rejection method and ABC-MCMC are provided as PJR-ABC and PJR-ABC-MCMC respectively. We analyze the error bound produced by PJR strategy. Our methodology establishes its practical value with desirable accuracy and efficiency. Finally, as a future direction, we plan to integrate PJR strategy with neural network as [24].

Acknowledgement. This study was funded by Scientific research fund of North University of China (No. XJJ201803).

References

1. Ahn, S., Korattikara, A., Welling, M.: Bayesian posterior sampling via stochastic gradient fisher scoring. In: Proceedings of the 29th International Conference on International Conference on Machine Learning, pp. 1771–1778 (2012)

2. Ahn, S., Shahbaba, B., Welling, M.: Distributed stochastic gradient MCMC. In: Proceedings of the 31st International Conference on Machine Learning (ICML-14), pp. 1044–1052 (2014)
3. Andrieu, C., Roberts, G.O.: The pseudo-marginal approach for efficient Monte Carlo computations. Ann. Stat. **37**, 697–725 (2009)
4. Barber, S., Voss, J., Webster, M., et al.: The rate of convergence for approximate Bayesian computation. Electron. J. Stat. **9**(1), 80–105 (2015)
5. Beaumont, M.A.: Approximate Bayesian computation in evolution and ecology. Annu. Rev. Ecol. Evol. Syst. **41**, 379–406 (2010)
6. Bernton, E., Jacob, P.E., Gerber, M., Robert, C.P.: Approximate Bayesian computation with the Wasserstein distance. J. Roy. Stat. Soc.: Ser. B (Stat. Methodol.) **81**(2), 235–269 (2019)
7. Biau, G., Cérou, F., Guyader, A., et al.: New insights into approximate Bayesian computation. In: Annales de l'Institut Henri Poincaré, Probabilités et Statistiques, vol. 51, pp. 376–403. Institut Henri Poincaré (2015)
8. Blum, M.G., François, O.: Non-linear regression models for approximate Bayesian computation. Stat. Comput. **20**(1), 63–73 (2010)
9. Blum, M.G., Nunes, M.A., Prangle, D., Sisson, S.A., et al.: A comparative review of dimension reduction methods in approximate Bayesian computation. Stat. Sci. **28**(2), 189–208 (2013)
10. Cabras, S., Nueda, M.E.C., Ruli, E., et al.: Approximate Bayesian computation by modelling summary statistics in a quasi-likelihood framework. Bayesian Anal. **10**(2), 411–439 (2015)
11. Chen, T., Fox, E., Guestrin, C.: Stochastic gradient Hamiltonian Monte Carlo. In: International Conference on Machine Learning, pp. 1683–1691 (2014)
12. Ding, N., Fang, Y., Babbush, R., Chen, C., Skeel, R.D., Neven, H.: Bayesian sampling using stochastic gradient thermostats. In: Advances in Neural Information Processing Systems, pp. 3203–3211 (2014)
13. Fearnhead, P., Prangle, D.: Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation. J. Roy. Stat. Soc.: Ser. B (Stat. Methodol.) **74**(3), 419–474 (2012)
14. Fu, T., Luo, L., Zhang, Z.: Quasi-Newton Hamiltonian Monte Carlo. In: Proceedings of the Thirty-Second Conference on Uncertainty in Artificial Intelligence, pp. 212–221 (2016)
15. Jiang, B., Wu, T.Y., Zheng, C., Wong, W.H.: Learning summary statistic for approximate Bayesian computation via deep neural network. Stat. Sin. **27**, 1595–1618 (2017)
16. Korattikara, A., Chen, Y., Welling, M.: Austerity in MCMC land: cutting the metropolis-hastings budget. In: International Conference on Machine Learning, pp. 181–189 (2014)
17. Lintusaari, J., Gutmann, M.U., Dutta, R., Kaski, S., Corander, J.: Fundamentals and recent developments in approximate Bayesian computation. Syst. Biol. **66**(1), e66–e82 (2017)
18. Ma, Y.A., Chen, T., Fox, E.: A complete recipe for stochastic gradient MCMC. In: Advances in Neural Information Processing Systems (2015)
19. Marin, J.M., Pudlo, P., Robert, C.P., Ryder, R.J.: Approximate Bayesian computational methods. Stat. Comput. **22**(6), 1167–1180 (2012)
20. Marjoram, P., Molitor, J., Plagnol, V., Tavaré, S.: Markov chain Monte Carlo without likelihoods. Proc. Natl. Acad. Sci. **100**(26), 15324–15328 (2003)
21. Meeds, E., Leenders, R., Welling, M.: Hamiltonian ABC. In: Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence, pp. 582–591 (2015)
22. Meeds, E., Welling, M.: GPS-ABC: Gaussian process surrogate approximate Bayesian computation. In: Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence, pp. 593–602 (2014)
23. Meeds, T., Welling, M.: Optimization Monte Carlo: efficient and embarrassingly parallel likelihood-free inference. In: Advances in Neural Information Processing Systems, pp. 2071–2079 (2015)

24. Mondal, M., Bertranpetit, J., Lao, O.: Approximate Bayesian computation with deep learning supports a third archaic introgression in Asia and Oceania. Nat. Commun. **10**(1), 246 (2019)
25. Pritchard, J.K., Seielstad, M.T., Perez-Lezaun, A., Feldman, M.W.: Population growth of human Y chromosomes: a study of Y chromosome microsatellites. Mol. Biol. Evol. **16**(12), 1791–1798 (1999)
26. Sisson, S.A., Fan, Y., Beaumont, M.: Handbook of Approximate Bayesian Computation. Chapman and Hall/CRC, New York (2018)
27. Sunnåker, M., Busetto, A.G., Numminen, E., Corander, J., Foll, M., Dessimoz, C.: Approximate Bayesian computation. PLoS Comput. Biol. **9**(1), e1002803 (2013)
28. Welling, M., Teh, Y.W.: Bayesian learning via stochastic gradient Langevin dynamics. In: Proceedings of the 28th International Conference on Machine Learning (ICML-11), pp. 681–688 (2011)
29. Wilkinson, R.: Accelerating ABC methods using Gaussian processes. In: Artificial Intelligence and Statistics, pp. 1015–1023 (2014)
30. Wilkinson, R.D.: Approximate Bayesian computation (ABC) gives exact results under the assumption of model error. Stat. Appl. Genet. Mol. Biol. **12**(2), 129–141 (2013)
31. Wood, S.N.: Statistical inference for noisy nonlinear ecological dynamic systems. Nature **466**(7310), 1102–1104 (2010)