# Kernel Recursive ABC: Point Estimation with Intractable Likelihood

**Takafumi Kajihara** [1 2] **Motonobu Kanagawa** [3] **Keisuke Yamazaki** [2] **Kenji Fukumizu** [4]

## Abstract

We propose a novel approach to parameter estimation for simulator-based statistical models with intractable likelihood. Our proposed method involves recursive application of kernel ABC and kernel herding to the same observed data. We provide a theoretical explanation regarding why the approach works, showing (for the population setting) that, under a certain assumption, point estimates obtained with this method converge to the true parameter, as recursion proceeds. We have conducted a variety of numerical experiments, including parameter estimation for a real-world pedestrian flow simulator, and show that in most cases our method outperforms existing approaches.

## 1. Introduction

Inference of parameters in a probabilistic model is an essential ingredient in model-based statistical approaches, both in the frequentist and Bayesian paradigms. Given a probabilistic model $P(y|\theta)$, which is a conditional distribution of observations $y$ given a parameter $\theta$, the aim is to make inference about the parameter $\theta^*$ that generated an observed data $y^*$. When the model $P(y|\theta)$ admits a conditional density $\ell(y|\theta)$, such an inference can be made on the basis of evaluations of $\ell(y^*|\theta)$; this is the *likelihood* of $y^*$ as a function of $\theta$. However, in modern scientific and engineering problems in which the model $P(y|\theta)$ is required to be sophisticated and complex, the likelihood function $\ell(y^*|\theta)$ might no longer be available. This may be because the density form of $P(y|\theta)$ is elusive, or the evaluation of the likelihood $\ell(y^*|\theta)$ is computationally very expensive. Such situations, in which $\ell(y|\theta)$ (or $P(y|\theta)$) are referred to as *intractable likelihood*, make the inference problem quite challenging and are commonly found in the literature on population genetics (Pritchard *et al.*, 1999) and dynamical systems (Toni *et al.*, 2009), to name just two.

*Approximate Bayesian Computation* (ABC) is a class of computational methods for Bayesian inference with intractable likelihood (Tavaré *et al.*, 1997; Pritchard *et al.*, 1999; Beaumont *et al.*, 2002) that is applicable as long as *sampling* from the model $P(y|\theta)$ is possible. Given a prior $\pi(\theta)$ on the parameter space, the basic ABC constructs a Monte Carlo approximation to the posterior $P_{y^*}(\theta) \propto P(y^*|\theta)\pi(\theta)$ in the following way: i) sample pairs $(y_i, \theta_i)$ of pseudo data $y_i$ and parameter $\theta_i$ from the joint distribution $P(y|\theta)\pi(\theta)$, where $i = 1, \dots, n$ for some $n \in \mathbb{N}$, ii) maintain only those parameters $\theta_i$ associated with $y_i$ that are "close enough" to the observed data $y^*$, and iii) regard them as samples from the posterior $P_{y^*}(\theta)$. ABC has been extensively studied in statistics and machine learning; see, e.g., Del Moral *et al.* (2012); Fukumizu *et al.* (2013); Meeds and Welling (2014); Park *et al.* (2016); Mitrovic *et al.* (2016).

In this paper, we rather take the frequentist perspective, and deal with the problem of *maximum likelihood estimation (MLE)* with intractable likelihood. That is, we consider situations in which one believes that there is a "true" parameter $\theta^*$ that generated the data $y^*$ and wishes to obtain a point estimate for it. This problem is also motivated by the following situations encountered in practice: 1) Consider a situation in which the model is computationally expensive (e.g., a state-space model) and one wants to perform prediction based on it. In this case fully Bayesian prediction would require simulation from each of sampled parameters, which might be quite costly. If one has a point estimate of the true parameter $\theta^*$, then the computational cost can be drastically reduced. 2) Consider a situation in which one only has limited knowledge w.r.t. model parameters. In this case, it is generally difficult to specify an appropriate prior distribution over the parameter space, and thus the resulting posterior may not be reliable.[1] Methods for point estimation with intractable likelihood have been reported in the literature, including the method of simulated-moments (McFadden, 1989), indirect inference (Gourieroux *et al.*,

[1]NEC Corporation [2]National Institute of Advanced Industrial Science and Technology [3]Max Planck Institute for Intelligent Systems [4]The Institute of Statistical Mathematics. Correspondence to: Takafumi Kajihara <t-kajihara@ct.jp.nec.com>.

*Proceedings of the* $35^{th}$ *International Conference on Machine Learning*, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

[1] For point estimation, one may think of using the maximum a posterior (MAP) estimate, but it may again be unreliable (as for the posterior distribution itself), if the prior distribution cannot be specified appropriately.

1993), ABC-based MAP estimation (Rubio *et al.*, 2013), noisy ABC-MLE (Dean *et al.*, 2014; Yıldırım *et al.*, 2015), an approach based on Bayesian optimization (Gutmann and Corander, 2016), and data-cloning ABC (Picchini and Anderson, 2017). We will discuss these existing approaches in Sec. 4.

Our contribution is in proposing a novel approach to point estimation with intractable likelihood on the basis of *kernel mean embedding of distributions* (Muandet *et al.*, 2017), a framework for statistical inference using reproducing kernel Hilbert spaces. Specifically, our approach extends *kernel ABC* (Fukumizu *et al.*, 2013; Nakagome *et al.*, 2013), a method for ABC using kernel embedding of conditional distributions (Song *et al.*, 2009; 2013), to point estimation with intractable likelihood. The novelty lies in combining kernel ABC with *kernel herding* (Chen *et al.*, 2010), a deterministic sampling method similar to quasi-Monte Carlo (Dick *et al.*, 2013), and in applying these two methods iteratively to the same observed data in a recursive way. We term this approach *kernel recursive ABC*. A theoretical explanation will be provided for this approach, discussing how such recursion yields a point estimate for the true parameter. We also discuss that the combination of kernel ABC and kernel herding leads to robustness against misspecification of a prior for the true parameter; this is an advantage over existing methods, and will be demonstrated experimentally.

This paper is organized as follows. We briefly review kernel ABC and kernel herding in Sec. 2 and propose kernel recursive ABC in Sec. 3. We report experimental results of comparisons with existing methods in Sec. 4. The experiments include parameter estimation for a real-world pedestrian flow simulator (Yamashita *et al.*, 2010), which may be of independent interest as application.

## 2. Background

### 2.1. Kernel ABC

Kernel ABC is an algorithm that executes ABC in a reproducing kernel Hilbert space (RKHS) and produces a reliable solution even in moderately large dimensional problems (Fukumizu *et al.*, 2013; Nakagome *et al.*, 2013). It is based on the framework of *kernel mean embeddings*, in which all probability measures are represented as elements in an RKHS (see Muandet *et al.* (2017) for a recent survey of this field). Let $\Theta$ be a measurable space, $k : \Theta \times \Theta \to \mathbb{R}$ be a measurable positive definite kernel, and $\mathcal{H}$ be its RKHS. In this framework, any probability measure $P$ on $\Theta$ will be represented as a Bochner integral

$$\mu_P := \int_\Theta k(\cdot, \theta) dP(\theta) \in \mathcal{H}, \tag{1}$$

which is called the *kernel mean* of $P$. If the mapping $P \to \mu_P$ is injective, in which case $\mu_P$ preserves all the information in $P$, the kernel $k$ is referred to as being *characteristic* (Fukumizu *et al.*, 2008). Characteristic kernels on $\Theta = \mathbb{R}^d$, for example, include Gaussian and Matérn kernels (Sriperumbudur *et al.*, 2010).

Let $\mathcal{Y}$ be another measurable space and assume that an observed data $y^* \in \mathcal{Y}$ is provided. ($y^*$ is often a set of sample points.) Given a conditional probability $P(y|\theta)$ and a prior $\pi(\theta)$, we wish to obtain the posterior distribution $P_{y^*}(\theta) \propto P(y^*|\theta)\pi(\theta)$.[2] As in a standard ABC, kernel ABC achieves this by first generating pairs of pseudo data and parameter $\{(y_i, \theta_i)\}_{i=1}^n$ from the joint distribution $P(y|\theta)\pi(\theta)$. It then estimates the kernel mean of the posterior $P_{y^*}$, which we denote by

$$\mu_{P_{y^*}} := \int_\Theta k(\cdot, \theta) dP_{y^*}(\theta) \in \mathcal{H}.$$

Given a measurable positive definite kernel $k_\mathcal{Y}$ on $\mathcal{Y}$, the estimator is given by

$$\hat{\mu}_{P_{y^*}} = \sum_{i=1}^n w_i k(\cdot, \theta_i) \in \mathcal{H}, \tag{2}$$

$$\boldsymbol{w} := (w_1, \ldots, w_n)^T := (G + n\delta I)^{-1} \boldsymbol{k}(y^*), \tag{3}$$

where $\boldsymbol{k}(y^*) := (k_\mathcal{Y}(y_1, y^*), \ldots, k_\mathcal{Y}(y_n, y^*))^T \in \mathbb{R}^n$, $G := (k_\mathcal{Y}(y_i, y_j))_{i,j=1}^n \in \mathbb{R}^{n \times n}$, $\delta > 0$ is a regularization constant, and $I \in \mathbb{R}^{n \times n}$ is an identity matrix. The estimator (2) is essentially an (RKHS-valued) kernel ridge regression (Grünewälder *et al.*, 2012): Given *training* data $\{(y_i, k(\cdot, \theta_i))\}_{i=1}^n$, the weights (3) provide an estimator for the mapping $y^* \Rightarrow k(\cdot, \theta^*)$. For consistency and convergence results, which require $\delta \to 0$ as $n \to \infty$, we re refer to Fukumizu *et al.* (2013) and Muandet *et al.* (2017).

### 2.2. Kernel herding

Kernel herding is a deterministic sampling technique based on the kernel mean representation of a distribution (Chen *et al.*, 2010) and can be seen as a greedy approach to quasi-Monte Carlo (Dick *et al.*, 2013). Consider sampling from $P$ using the kernel mean $\mu_P$ (1), and assume that one is able to evaluate function values of $\mu_P$. Kernel herding greedily obtains sample points $\theta_1, \theta_2, \ldots, \theta_n$ by iterating the following steps: Defining $h_0 := \mu_P$,

$$\theta_{t+1} = \operatorname*{argmax}_{\theta \in \Theta} h_t(\theta), \tag{4}$$

$$h_{t+1} = h_t + \mu_P - k(\cdot, \theta_{t+1}) \in \mathcal{H}, \tag{5}$$

where $t = 0, \ldots, n-1$. Chen *et al.* (2010) has shown that, if there exists a constant $C > 0$ such that $k(\theta, \theta) = C$ for all $\theta \in \Theta$, this procedure will be identical to the greedy

[2] There is abuse of notation here, as $P(y|\theta)$ does not denote a conditional density but a conditional distribution.

**Algorithm 1** Kernel Recursive ABC

```
Input: A prior distribution $\pi$, an observed data $y^*$, a data generator $P(y|\theta)$, the number $N_{\rm iter}$ of iterations, the number $n$ of simulated pairs, a kernel $k$ on $\Theta$, a kernel $k_{\mathcal{Y}}$ on $\mathcal{Y}$, and a regularization constant $\delta > 0$
Output: A point estimate $\acute{\theta}$.
for $N = 1, ..., N_{\rm iter}$ do
  if $N = 1$ then
    for $i = 1, ..., n$ do
      Sample $\theta_{1,i} \sim \pi(\theta)$ i.i.d.
    end for
  end if
  for $i = 1, ..., n$ do
    Generate $y_{N,i} \sim P(\cdot|\theta_{N,i})$
  end for
  Compute $G := (k_{\mathcal{Y}}(y_{N,i}, y_{N,i}))_{i,j=1}^{n} \in \mathbb{R}^{n \times n}$ and $\boldsymbol{k}(y) := (k_{\mathcal{Y}}(y_{N,i}, y^*))_{i=1}^{n} \in \mathbb{R}^n$.
  Calculate $\boldsymbol{w} = (w_1, \dots, w_n)^T \in \mathbb{R}^n$ by Eq.(3).
  Construct a kernel mean estimate of the powered posterior $\hat{\mu}_{P_N} := \sum_{i=1}^{n} w_i k(\cdot, \theta_{N,i})$
  Sample $\{\theta_{N+1,t}\}_{t=1}^{n}$ by performing kernel herding Eqs.(4) (5) with $\mu_P := \hat{\mu}_{P_N}$.
end for
Obtain a point estimate $\acute{\theta} := \theta_{N_{\rm iter}+1,1}$
```

minimization of the maximum mean discrepancy (MMD) (Gretton *et al.*, 2007; 2012):

$$\epsilon_n := \left\| \mu_P - \frac{1}{n} \sum_{t=1}^{n} k(\cdot, \theta_t) \right\|_{\mathcal{H}}, \tag{6}$$

where $\| \cdot \|_{\mathcal{H}}$ denotes the norm of $\mathcal{H}$. That is, the points $\theta_1, \dots, \theta_n$ are obtained so as to (greedily) minimize the distance $\varepsilon_n$ between $\mu_P$ and the empirical kernel mean $\frac{1}{n} \sum_{t=1}^{n} k(\cdot, \theta_t)$. The generated points $\theta_1, \dots, \theta_n$ are also called super-samples because they are more informative than those from random sampling; this is in the sense that error decreases at the rate $\epsilon_n = O(n^{-1})$ if the RKHS is finite-dimensional (Bach *et al.*, 2012), which is faster than the rate $\epsilon_n = O(n^{-1/2})$ of random sampling (Smola *et al.*, 2007). Convergence guarantees are also provided even when the optimization problem in (4) is solved approximately (Lacoste-Julien *et al.*, 2015) and when the kernel mean $\mu_P$ is replaced by an empirical estimate $\hat{\mu}_P$ of the form (2) (Kanagawa *et al.*, 2016b). Note that the decay $\epsilon_n \to 0$ of the error (6) as $n \to \infty$ implies the convergence of expectation $\frac{1}{n} \sum_{t=1}^{n} f(\theta_t) \to \int f(x) dP(x)$ for all functions $f$ in the RKHS $\mathcal{H}$ and for functions $f$ that can be approximated well by the RKHS functions (Kanagawa *et al.*, 2016a).

## 3. Proposed method

Our idea is to recursively apply Bayes' rule to the same observed data $y^*$ by using the posterior obtained in one iteration as a prior for the next iteration. For this, let $\ell(\theta) := \ell(y^*|\theta)$ be a likelihood function and $\pi(\theta)$ be a prior density, where $\theta \in \Theta$, with $\Theta$ being a measurable space. Consider the population setting in which no estimation procedure is involved. After the $N$-th recursion, the posterior distribution becomes

$$p_N(\theta) := C_N^{-1}\, \pi(\theta) (\ell(\theta))^N, \tag{7}$$

where $C_N := \int_\Theta \pi(\theta) \left(\ell(\theta)\right)^N d\theta$ is a normalization constant. We refer here to this as a *powered posterior*. If $\ell$ has a unique global maximum at $\theta_\infty \in \Theta$ and the support of $\pi$ contains $\theta_\infty$, one can show that $p_N$ converges weakly to the Dirac distribution $\delta_{\theta_\infty}$ at $\theta_\infty$ under certain conditions (Lele *et al.*, 2010). In other words, the effect of the prior diminishes as the recursion proceeds, and the powered posterior degenerates at the maximum likelihood point, providing a method for MLE. A similar idea has been discussed by Doucet *et al.* (2002); Lele *et al.* (2010) in the context of data augmentation and data cloning, in which one replicates the observed data $y^*$ multiple times and applies Bayes' rule once; our approach is different, as we employ recursive applications of Bayes' rule multiple times (this turns out to be beneficial in our approach, as is shown below).

Based on the above idea, we propose to recursively applying kernel ABC (Sec. 2.1) and kernel herding (Sec. 2.2). Specifically, the proposed method (Algorithm 1) iterates the following procedures: (i) At the $N$-th iteration, the kernel mean $\mu_{P_N} := \int k(\cdot, \theta) p_N(\theta) d\theta$ of the powered posterior (7) is estimated using simulated pairs $\{(\theta_{N,i}, y_{N,i})\}_{i=1}^{n}$ via kernel ABC; (ii) from the estimate $\hat{\mu}_{P_N}$ of $\mu_{P_N}$ given in (i), new parameters $\{\theta_{N+1,i}\}_{i=1}^{n}$ are generated via kernel herding, and new pseudo-data $\{y_{N+1,i}\}_{i=1}^{n}$ are generated from the simulator $P(y_{N+1,i}|\theta_{N+1,i})$ in the $N+1$-th iteration. After iterating these procedures $N_{\rm iter}$ times, point estimate $\acute{\theta}$ for the true parameter is given as the first point $\theta_{N_{\rm iter}+1,1}$ from kernel herding at the last iteration.

**Auto-correction mechanism.** An interesting feature of the proposed approach is that, as experimentally indicated in Sec. 4.2, it is equipped with an auto-correction mechanism: If the parameters $\theta_{N,1}, \dots, \theta_{N,n}$ at the $N$-th iteration are far apart from the true parameter $\theta^*$, then Algorithm 1 searches for the parameters $\theta_{N+1,1}, \dots, \theta_{N+1,n}$ at the next iteration, so as to explore the parameter space $\Theta$. For instance, if the prior $\pi(\theta)$ is misspecified, meaning that the true parameter $\theta^*$ is not contained in the support of $\pi(\theta)$, then the initial parameters $\theta_{1,1}, \dots, \theta_{1,n}$ from $\pi(\theta)$ are likely to be apart from the true parameter $\theta^*$. The auto-correction mechanism makes the proposed method robust to such misspecification and makes it suitable for use in situations in which one lacks appropriate prior knowledge about the true parameter.

To explain how this works, let us explicitly write down the procedure (4) (5) of kernel herding as used in Algorithm 1.

Given that $t\ (< n)$ points $\theta_{N+1,1}, \ldots, \theta_{N+1,t}$ have already been generated, the next point $\theta_{N+1,t+1}$ is obtained as

$$\theta_{N+1,t+1} := \tag{8}$$
$$\operatorname*{argmax}_{\theta \in \Theta} \sum_{i=1}^{n} w_i k(\theta, \theta_{N,i}) - \frac{1}{t+1} \sum_{i=1}^{t} k(\theta, \theta_{N+1,i}),$$

where the weights $w_1, \ldots, w_n$ are given as (3). Assume that all the simulated parameters $\theta_{N,1}, \ldots, \theta_{N,n}$ at the $N$-th iteration are far apart from the true parameter $\theta^*$: If $N = 1$, these are the parameters sampled from the prior $\pi(\theta)$. Then it is likely the resulting simulated data $y_{N,1}, \ldots, y_{N,n}$ are dissimilar to the observed data $y^*$. In this case, each component of the vector $\boldsymbol{k}(y) := (k_{\mathcal{Y}}(y_{N,i}, y^*))_{i=1}^n \in \mathbb{R}^n$ becomes nearly 0, since $k_{\mathcal{Y}}(y_{N,i}, y^*)$ quantifies the similarity between $y^*$ and $y_{N,i}$. As a result, each of the weights $w_1, \ldots, w_n$ given by kernel ABC (3) also become nearly 0, and thus the first term on the right side in (8) will be ignorable. The point $\theta_{N+1,t+1}$ is then obtained so as to roughly maximize the second term $-\frac{1}{t+1}\sum_{i=1}^t k(\theta_{N+1,t+1}, \theta_{N+1,i})$, or, equivalently, so as to minimize $\sum_{i=1}^t k(\theta_{N+1,t+1}, \theta_{N+1,i})$. Since the kernel $k(\theta_{N+1,t+1}, \theta_{N+1,i})$ measures the similarity between $\theta_{N+1,t+1}$ and $\theta_{N+1,i}$, the new point $\theta_{N+1,t+1}$ is located apart from the points $\theta_{N+1,1}, \ldots, \theta_{N+1,t}$ generated so far. In this way, the parameters $\theta_{N+1,1}, \ldots, \theta_{N+1,n}$ at the $N+1$-th iteration are made to explore the parameter space $\Theta$ if parameters $\theta_{N,1}, \ldots, \theta_{N,n}$ at the $N$-th iteration are far apart from the true parameter $\theta^*$.

### 3.1. Theoretical analysis

We provide here a theoretical basis for the proposed recursive approach. Since the consistency of kernel ABC and kernel herding have already been established in the literature (Fukumizu *et al.*, 2013; Bach *et al.*, 2012), we focus on convergence analysis in the population setting, that is, convergence analysis for the kernel mean of the powered posterior (7) and for the resulting point estimate. We nevertheless note that convergence analysis of the overall procedure of Algorithm 1 remains an important topic for future research. All the proofs can be found in the Supplementary Materials.

Below, we let $\Theta$ be a Borel measurable set in $\mathbb{R}^d$. Denote by $P_N$ the probability measure induced by the powered posterior density $p_N$ (7), and let $\mu_{P_N} := \int k(\cdot, \theta) dP_N(\theta) \in \mathcal{H}$ be its kernel mean, where $k$ is a kernel on $\Theta$ and $\mathcal{H}$ is its RKHS. We require the following assumption for the likelihood function $\ell$ and the prior $\pi$ for theoretical analysis.

**Assumption 1.** *(i) $\ell$ has a unique global maximum at $\theta_\infty \in \Theta$, and $\pi(\theta_\infty) > 0$; (ii) $\pi$ is continuous at $\theta_\infty$, $\ell$ has continuous second derivatives in the neighborhood of $\theta_\infty$, and the Hessian of $\ell$ at $\theta_\infty$ is strictly negative-definite.*

Our first result below shows that, under Assumption 1, the powered posterior $P_N$ (7) converges to the Dirac distribution $\delta_{\theta_\infty}$ in the RKHS $\mathcal{H}$ as $N \to \infty$; this provides a theoretical basis for recursively applying the kernel ABC.

**Proposition 1.** *Let $\Theta \subset \mathbb{R}^d$ be a Borel measurable set and $k : \Theta \times \Theta \to \mathbb{R}$ be a continuous bounded kernel. Under Assumption 1, we have $\lim_{N\to\infty} \|\mu_{P_N} - k(\cdot, \theta_\infty)\|_{\mathcal{H}} = 0$.*

Proposition 2 below provides a justification for the use of the first point of kernel herding (here this is $\theta_N := \operatorname{argmin}_{\tilde{\theta}\in\Theta} \|\mu_{P_N} - k(\cdot, \tilde{\theta})\|_{\mathcal{H}}$; see Sec. 2.2) as a point estimate of $\theta_\infty$. To this end, we introduce the following assumption on the kernel, which is satisfied by, for example, Gaussian and Matérn kernels.

**Assumption 2.** *(i) There exists a constant $C > 0$ such that $k(\theta, \theta) = C$ for all $\theta \in \Theta$. (ii) It holds that $k(\theta, \theta') < C$ for all $\theta, \theta' \in \Theta$ with $\theta \neq \theta'$.*

**Proposition 2.** *Let $\Theta \subset \mathbb{R}^d$ be a compact set, and $k : \Theta \times \Theta \to \mathbb{R}$ be a continuous, bounded kernel. Let $\theta_N := \operatorname{argmin}_{\tilde{\theta}\in\Theta} \left\|\mu_{P_N} - k(\cdot, \tilde{\theta})\right\|_{\mathcal{H}}$. If Assumptions 1 and 2 hold, then we have $\theta_N \to \theta_\infty$ as $N \to \infty$.*

We make a few remarks regarding Assumption 1. The assumption that $\ell$ has a unique global maximum is not satisfied if the model is singular, an example being mixture models: In this case there are multiple global maximums. However, our experiment in Sec. 4.5 shows that even for mixture models, the proposed method works reasonably well. This suggests that, in an empirical setting, a point estimate may converge to one of the global maximums. The assumption $\pi(\theta_\infty) > 0$ will also not be satisfied if the support $\pi$ does not contain $\theta_\infty$, but the proposed method performs well even in this case (as shown in 4.2), possibly thanks to the auto-correction mechanism explained above. We reserve further analysis of these properties for future work.

## 4. Experiments

We have conducted a variety of experiments comparing the proposed method with existing approaches. We begin with a quick review of these approaches (Sec. 4.1), and report experimental results on point estimation with a misspecified prior (Sec. 4.2), population dynamics of the blowfly (Sec. 4.3), alpha stable distributions (Sec. 4.4), Gaussian mixture models with redundant components (Sec. 4.5), and a real-world pedestrian simulator (Sec. 4.6).

### 4.1. Existing approaches and experimental settings

**K2-ABC** (Park *et al.*, 2016) is an ABC method that represents the empirical distributions of simulated and test observations as kernel means in an RKHS. For each of simulated parameters, the associated weight is calculated by using the RKHS distance between the kernel means (i.e., MMD), and the resulting weighted sample is treated as a posterior distribution. **Adaptive SMC-ABC** (Del Moral *et al.*, 2012) is a rejection-based approach based on sequential Monte Carlo, which sequentially updates the tolerance level and the associated proposal distribution in an adaptive manner. This method is a state-of-the-art ABC approach. The approach by Gutmann and Corander (2016), which we refer to as **Bayesian Optimization** for simplicity, is a method for MLE with intractable likelihood based on Bayesian optimization (Brochu *et al.*, 2010). This method optimizes the parameters in a intractable model so as to minimize the discrepancy between the simulated and test observations. Note that comparison with this method in terms of computation time may not make sense (although we report them for purposes of completeness), as we used publicly available code[3] for implementation. **The method of simulated moments (MSM)** (McFadden, 1989) optimizes the parameter in the model so that the resulting moments of simulated data match those of observe data. MSM may be seen a special case of indirect inference (Gourieroux *et al.*, 1993), an approach studied in econometrics.[4] **Data-cloning ABC (ABC-DC)** (Picchini and Anderson, 2017) is an approach combining ABC-MCMC (Marjoram *et al.*, 2003) and Data Cloning (Lele *et al.*, 2010), replicating observed data multiple times to achieve MLE with intractable likelihood.

**Experimental settings.** Unless otherwise specified, the following settings were applied in the experiments. For all the methods that employed kernels, we used Gaussian kernels. The discrepancy between the simulated and observed data was measured by the *energy distance* (Székely and Rizzo, 2013), which is a standard metric for distributions in statistics and can be computed only from pairwise Euclidean distances between data points. Since the usual quadratic time estimator was too costly, we used a linear time estimator for computing the energy distance (see the Supplementary Materials for details).

For each method, unless otherwise specified, we determined the hyper-parameters on the basis of the cross-validation-like approach described in Park *et al.* (2016, Sec. 4). That is, to evaluate one configuration of hyper-parameters, we first used $75\%$ of the observed data for point estimation and then computed the discrepancy between the rest of the observed data and the ones simulated from point estimates; after applying this procedure to all candidate configurations, the one with the lowest discrepancy was finally selected. The bandwidth of a Gaussian kernel was selected from candidate values, each of which is the median (of pairwise distances) multiplied by logarithmically equally spaced values between $2^{-4}$ and $2^4$ (Takeuchi *et al.*, 2006, Sec. 5.1.1). Regularization constants for the proposed method and kernel ABC, as well as the soft threshold for K2-ABC, were selected from logarithmically spaced values between $10^{-4}$ and 1. To compute MMD for K2-ABC, a linear time estimator (Gretton *et al.*, 2012, Sec. 6) was used to reduce computational time, as the usual quadratic time estimator was too costly. For Adaptive SMC-ABC, the initial tolerance level was set as the median of pairwise distances between the observed and simulated data. For Bayesian Optimization, we used Expected Improvement as an acquisition function, and all the hyper-parameters were marginalized out following the approach of Snoek *et al.* (2012, Sec. 3.2). For MSM, the number of moments were selected from a range up to 30 by the cross-validation like approach. For ABC-DC, we employed, in particular, dynamic ABC-DC, which automatically adjusts its associated parameters. To obtain point estimates with kernel ABC and K2-ABC, we computed the means of the resulting posterior distributions. For Adaptive SMC-ABC, point estimates were obtained as posterior means as well as MAP estimates by applying the mean shift algorithm to posterior weighted samples (Fukunaga and Hostetler, 1975), the latter essentially being an approach suggested by Rubio *et al.* (2013).

The following abbreviations may be used for the sake of simplicity; kernel recursive ABC is referred to as **KR-ABC**, kernel ABC as **K-ABC**, adaptive SMC-ABC as **SMC-ABC**, Bayesian Optimization as **BO**, and Dynamic ABC-DC as **ABC-DC**. For our method, we also report results based on a half number of iterations, which we call **KR-ABC (less)**.

### 4.2. Multivariate Gaussian distribution with a severely misspecified prior

As a proof of concept regarding the auto-correction mechanism of the proposed method described in Sec.3, we have performed an experiment for when the prior distribution is severely misspecified (see the Supplementary Materials for an illustration). The task is to estimate the mean vector of a 20-dimensional Gaussian distribution $\mathrm{Normal}(\mu, \Sigma)$, where the true mean vector is $\mu := (10, 50, 90, 130, 180, 280, 390, 430, 520, 630, 1010, 1050, 1090, 1130, 1180, 1280, 1390, 1430, 1520, 1630)^T \in \mathbb{R}^{20}$. The covariance matrix $\Sigma \in \mathbb{R}^{20\times 20}$ is assumed to be known and is a diagonal matrix with all diagonal elements being $40$. Test data $y^*$ consisted of 100 i.i.d. observations from this Gaussian distribution. As a prior for the mean vector $\mu$, we used the uniform distribution on $[9 \times 10^6, 10^7]^{20}$, which is extremely misspecified. For Bayesian optimization, the space to be explored was set as $[0, 10^7]^{20}$.

In this experiment, each pseudo data was made from 100 observations simulated with one parameter configuration. K2-ABC and K-ABC used 3000 pairs of a parameter and pseudo-data. For the proposed method and SMC-ABC, we generated 100 pairs of a parameter and pseudo-data for the initial iteration, and then the iterations were repeated 30

[3] https://sheffieldml.github.io/GPyOpt/

[4] MSM is a special case of indirect inference because the moments can be regarded as the parameters of an auxiliary model.

*Table 1.* Results for multivariate Gaussian distributions in Sec. 4.2

| Algorithm | parameter error | data error | cputime |
|---|---|---|---|
| KR-ABC | 0.70(0.29) | 0.008(0.004) | 866.02(26.12) |
| KR-ABC (less) | 7.22(3.28) | 0.02(0.24) | 353.498(23.05) |
| K2-ABC | >1e+6 (>1e+3) | >1e+5 (>1e+3) | 209.51(11.49) |
| K-ABC | >1e+6 (>1e+3) | >1e+5 (>1e+3)) | 403.93(24.97) |
| SMC-ABC (mean) | >1e+6 (>1e+3) | >1e+5 (>1e+3) | 590.41(29.54) |
| SMC-ABC (MAP) | >1e+6 (>1e+3) | >1e+5 (>1e+3) | 590.41(29.54) |
| ABC-DC | >1e+6 (>1e+3) | >1e+5 (>1e+3) | 313.99(16.85) |
| BO | >1e+5(>1e+4) | >1e+5 (>1e+4) | 25940.86(936.40) |
| MSM | >1e+5(>1e+4) | >1e+5(>1e+4) | 307.42(67.94) |

*Table 2.* Results for blowfly population dynamics in Sec.4.3

| Algorithm | parameter error | data error | cputime |
|---|---|---|---|
| KR-ABC | 0.47(0.11) | 43.85(37.24) | 101.143(13.25) |
| KR-ABC (less) | 0.57(0.21) | 67.57(47.11) | 32.98(1.21) |
| K2-ABC | 0.81(0.42) | 67.45(77.86) | 23.47(1.59) |
| K-ABC | 0.62(0.09) | 89.37(29.22) | 30.66(2.57) |
| SMC-ABC (mean) | 0.83 (0.10) | 170.41 (47.91) | 38.50(2.34) |
| SMC-ABC (MAP) | 0.84 (0.12) | 163.19(42.51) | 38.50(2.34) |
| ABC-DC | 0.89(0.17) | 134.12(58.92) | 29.94(4.57) |
| BO | 0.70(0.28) | 108.18(67.08) | 3217.40(157.31) |
| MSM | 0.67(0.08) | 89.17(33.20) | 25.46(8.26) |

times, resulting in a total of 3000 simulations. For the proposed method, the bandwidth of the kernel $k_{\mathcal{Y}}$ on observed data was recomputed for each iteration, using the median heuristic. For SMC-ABC, the parameter $\alpha \in (0, 1)$, which controls the trade-off between the speed of convergence and the accuracy of posterior approximation, was set to be 0.3, as we found this value to be the best in terms of the trade-off.

For each method, we ran 30 independent trials, and the results in averages and standard deviations are shown in Table 1, where the *parameter error* is the mean (over 20 dimensions) of the absolute difference between the estimated and the true parameter values divided by the true value, and the *data error* is the energy distance between the true data and pseudo data simulated with the estimated parameter. Surprisingly, the proposed method successfully approached the true parameter even when the prior was severely misspecified. As discussed in Sec 3 and demonstrated in the Supplementary Materials, this would appear to be because of the use of kernel herding, which automatically widens the space to explore when simulated data is far apart from test data. As expected, other methods were unable to approach the true parameter.

### 4.3. Ecological dynamic systems: blowfly

Following Park *et al.* (2016), we performed an experiment on parameter estimation with a dynamical system of blowfly populations (Wood, 2010), which is defined as

$$N_{t+1} = P N_{t-\tau} \exp\left(-N_{t-\tau}/N_0\right) e_t + N_t \exp(-\delta \epsilon_t),$$

where $t = 1, \ldots, T$ are time indices, $N_t$ is the population at time $t$, $e_t \sim \mathrm{Gam}(\frac{1}{\sigma_p^2}\sigma_p^2)$ and $\epsilon_t \sim \mathrm{Gam}(\frac{1}{\sigma_d^2}, \sigma_d^2)$ are independent Gamma-distributed noise, and $\theta := (P \in \mathbb{N},\ N_0 \in \mathbb{N},\ \sigma_d \in \mathbb{R}_+,\ \sigma_p \in \mathbb{R}_+,\ \tau \in \mathbb{N},\ \delta\ \in \mathbb{R}_+)$ are the parameters of the system. The task is to estimate $\theta$ from observed values of $N_1, \ldots, N_T$. We set the true parameters as $\theta = (29, 260, 0.6, 0.3, 7, 0.2)$, and the time-length $T$ for both the observed and pseudo data as $T = 1000$. Following Park *et al.* (2016, Sec. 4), for each parameter we defined a Gaussian prior on its logarithm (see the Supplementary Materials for a definition). In this experiment, for all methods we converted the observed and pseudo-data into histograms with 1000 bins (i.e., we treated each data as a 1000 dim. vector), as this produced better results. K2-ABC and K-ABC used 1300 pairs of a parameter and a pseudo-data item. For the proposed method and SMC-ABC, we generated 100 pairs of a parameter and pseudo-data for the initial iteration, and the iterations were then repeated 13 times, resulting in total 1300 simulations. For SMC-ABC, we set the parameter $\alpha \in (0, 1)$ to be 0.3, as in Sec. 4.2.

For each method we performed 30 independent trials, and the results are summarized in Table 2. The proposed method performed the best, even when the number of simulations was halved (i.e., KR-ABC (less)).

### 4.4. Multivariate elliptically contoured alpha stable distribution

In addition to the competitive methods described earlier, we performed a comparison with the method called *noisy ABC-MLE* (Yıldırım *et al.*, 2015). This method assumes that sampling from the intractable model can be realized by deterministic mapping applied to a simple random variable, and that the gradient of the deterministic mapping is available. This method can, then, only be applied to a limited class of generative models, though for such models it can perform well. Although the main scope of this paper is on simulation models in which such gradient information is unavailable, we performed this experiment in order to see how the proposed method compared with this method without relying on the gradient information.

We also considered parameter estimation with *multivariate elliptically contoured alpha stable distributions* (Nolan, 2013), which subsume heavy-tailed and skewed distributions and are popular for modeling financial data. This family of distributions in general does not admit closed-form expressions for density functions, which means they are "intractable" in the sense that the standard procedure for parameter estimation cannot be employed. However, sampling of a random vector $\boldsymbol{X} \in \mathbb{R}^d$ from this family is possible in the following way:

$$\begin{aligned}
\boldsymbol{X} &:= A^{1/2}\boldsymbol{G} + \delta \in \mathbb{R}^d, \quad \boldsymbol{G} \sim \mathrm{Normal}(\mathbf{0}, Q), \\
A &:= \tau_\theta(U_1, U_2) \in \mathbb{R}, \\
&\phantom{:=}\ U_1 \sim \mathrm{Unif}(-\pi/2, \pi/2), \quad U_2 \sim \mathrm{Exp}(1),
\end{aligned}$$

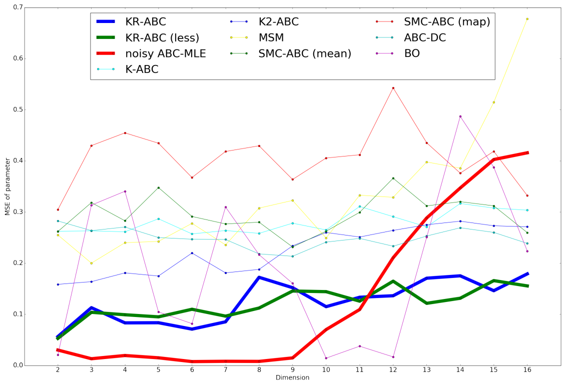

*Figure 1.* Results for multivariate elliptically contoured alpha stable distributions in Sec. 4.4. For each dimension (vertical axis), the averages of mean squared errors (MSE) over 30 trials are shown.

where $Q \in \mathbb{R}^{d \times d}$ is positive definite, $\delta \in \mathbb{R}^d$, $\theta := (\alpha, \beta, \mu, \sigma) \in (0, 2] \times [-1, 1] \times \mathbb{R} \times [0, \infty)$, $\tau_\theta$ is a deterministic mapping whose concrete form is described in the Supplementary Materials, and Unif and Exp denote uniform and exponential distributions, respectively.

We dealt with estimation of $\alpha := 1.3$ and $Q$, while fixing the other parameters as $\delta := \mathbf{0}$, $\beta := 1$, $\mu := 0$, and $\sigma := 1$. We restricted $Q$ to be a positive definite matrix such that all diagonal elements are the same and so are the off-diagonal elements. We defined true $Q$ to be a matrix whose diagonal elements are $1.0$ and off-diagonals are $0.2$. Therefore the task was to estimate these three values (i.e., 1.3 for $\alpha$, and 1.0 and 0.2 for $Q$). We used $\mathrm{Unif}[0, 2]$ as a prior for $\alpha$, and $\mathrm{Unif}[0, 5]$ as a prior for each of the diagonal and off-diagonal values of $Q$.

Figure 1 shows results for the averages of mean square errors in parameter estimation over 30 independent trials, with variation in the dimensionality $d$ from 2 to 16. For each method we sampled a total of 1400 pairs of a parameter and a pseudo-data item, and for iterative methods we used 100 pairs in each iteration. Each pseudo-data (and observed-data) was made up of 1000 points. The noisy ABC-MLE exploited the gradient information in $\tau_\theta$, while the other methods did not. The proposed method was competitive with BO and outperformed the other methods with the exception of the noisy ABC-MLE. Although the noisy ABC-MLE was accurate for lower-dimensionality (as expected), it exhibited a steep increase in errors for higher dimensionality. In contrast, the performance degradation of the proposed method was mild for higher dimensionality.

### 4.5. Gaussian mixture with redundant components

We consider here a parametric model in which there exist redundant parameters for expressing given data. We are interested in whether point estimation with the proposed method results in elimination of the redundant parameters when applied to such a model. This was motivated by Yamazaki and Kaji (2013), who argued that, for mixture models, the use of a Dirichlet prior with a sufficiently small concentration parameter leads to elimination of unnecessary components. We therefore focus on mixture models with redundant components.

*Table 3.* Results for the Gaussian mixture model in Sec. 4.5

| Algorithm | $\phi$ error | $\mu$ error | data error | cputime |
|---|---|---|---|---|
| KR-ABC | 0.159(0.106) | 54.14(8.71) | 0.03(0.05) | 281.59(12.55) |
| KR-ABC (less) | 0.22(0.13) | 64.04(17.58) | 0.11(0.15) | 147.39(16.76) |
| K2-ABC | 0.53(0.02) | 93.98(3.84) | 0.69(0.10) | 89.43(7.56) |
| K-ABC | 0.50(0.02) | 92.57(12.77) | 0.67(0.17) | 135.01(11.35) |
| SMC-ABC (mean) | 0.51(0.04) | 83.19(27.86) | 0.23(0.15) | 214.35(8.77) |
| SMC-ABC (MAP) | 0.21(0.13) | 72.76(56.24) | 0.12(0.08) | 214.35(8.77) |
| ABC-DC | 0.48(0.16) | 137.84(50.06) | 0.43(0.14) | 149.74(21.22) |
| BO | 0.37(0.08) | 80.79(36.17) | 0.82(0.68) | 13775.48(1438.9) |
| MSM | 0.24(0.08) | 117.02(11.48) | 0.24(0.08) | 171.58(59.38) |

Specifically, we considered Gaussian mixture models. We defined the true model as a *two-component* Gaussian mixture $\sum_{i=1}^{2} \phi_i \mathrm{Normal}(\mu_\mathrm{i}, 20)$ of equal variances. The task was to estimate the mixture coefficients $(\phi_1, \phi_2, ) := (0.7, 0.3)$ and the associate means $(\mu_1, \mu_2) := (110, 70)$, provided 3000 i.i.d. sample points from the model as observed data $y^*$. We employed an over-parametrized model for point estimation (i.e., no method used the knowledge that the truth consisted of 2 components), which is a *four-component* Gaussian mixture $\sum_{i=1}^{4} \phi_i \mathrm{Normal}(\mu_\mathrm{i}, 20)$. We used a 4-dimensional Dirichlet distribution with equal concentration parameters $0.01$ as a prior for the coefficients $(\phi_1, \ldots, \phi_4)$, and $\mathrm{Normal}(0, 100)$ as a prior for each of $\mu_1, \ldots, \mu_4$.

For each method, we generated a total of 1000 pairs of a parameter and pseudo-data, and for iterative methods, we made use of 100 pairs in each iteration, resulting in 10 iterations. Each pseudo-data consisted of 3000 simulated observations. For all the methods, we converted each data item into a histogram of 300 bins and treated it as a 300 dim. vector since this resulted in better performances. We set the parameter $\alpha \in (0, 1)$ of SMC-ABC to be 0.2, as this performed well in this experiment.

We ran each algorithm 30 times, and the resulting average errors and standard deviations are shown in Table 3, where the $\phi$ error and $\mu$ error denote the errors for the coefficients and the means, respectively, as measured in terms of Euclidean distance. More precisely, since any permutation of component labels will result in the same model, we first sorted the estimated parameters $\{(\phi_i, \mu_i)\}$ so that $\phi_1 \geq \cdots \geq \phi_4$, and we then measured the errors w.r.t. the ground truth $\phi := (0.7, 0.3, 0, 0)$ and $\mu := (110, 70)$. For the $\mu$ error, we computed the errors only for the estimated means $\mu_1, \mu_2$ associated with the two largest coefficients since there was no ground truth for the redundant components $\mu_3, \mu_4$. Results show that the proposed KR-ABC performed best, indicating that the dominant components were successfully estimated.

### 4.6. Real-world pedestrian simulator

Our final experiment was parameter estimation with *CrowdWalk*, a publicly available real-world simulator[5] for the movements of pedestrians in a commercial district (Yamashita *et al.*, 2010). It has been used to gain insights into pedestrian behavior at a variety of events and occurrences, such as fireworks festivals and evacuations after earthquakes. As this simulator is complicated and also computationally expensive, its likelihood function is intractable.

Using CrowdWalk, we simulated the movements of pedestrians in Ginza, a commercial district in Tokyo (see Supplementary Materials for an illustration). Specifically, we modeled pedestrians as a mixture of multiple groups, each of which has the following 6 parameters (below $i$ denotes the index of a group): (1) $\theta_i^{(N)} \in \mathbb{N}$: the number of pedestrians in the group; (2) $\theta_i^{(T)} \in \mathbb{R}_+$: the time when the group starts to move; (3) $\theta_i^{(S)} \in \mathbb{R}^2$: the starting location of the group (e.g., stations); (4) $\theta_i^{(G)} \in \mathbb{R}^2$: the goal location of the group; (5) $\theta_i^{(P)} \in \mathbb{R}^2$: the intermediate location(s) that the pedestrians in the group visit (e.g., stores); and (6) $\theta_i^{(R)} \in \mathbb{R}_+$: the time duration(s) of the pedestrians' visit(s) at the intermediate location(s).

In this experiment, we focused on estimation of the first two parameters $\theta_i^{(N)}$, $\theta_i^{(T)}$, and fixed the other parameters. We defined the true model as a mixture of 5 pedestrian groups, and set their parameters as $(\theta_1^{*(N)}, \dots, \theta_5^{*(N)}) := (100, 100, 100, 100, 100)$ and $(\theta_1^{*(T)}, \dots, \theta_5^{*(T)}) := (30, 60, 90, 120, 150)$. As in Sec. 4.5, we used a redundant model of a mixture of 10 groups for parameter estimation. The goal was to detect the active 5 groups of the true model, without knowing that the truth consists of 5 groups. For simplicity, 5 (unknown) groups among the 10 candidate groups included the parameters of the true model other than $\theta_i^{*(N)}$, $\theta_i^{*(T)}$; see the Supplementary Materials for details.

We defined prior distributions as follows. First we assumed the total number 500 of pedestrians to be known. The mixing coefficients of the mixture of 10 groups are given by $(\phi_1, \dots, \phi_{10}) = (\theta_1^{(N)}, \dots, \theta_{10}^{(N)})/500$. Thus, rather than directly putting a prior on $(\theta_1^{(N)}, \dots, \theta_{10}^{(N)})$, we defined a prior on the mixing coefficients $(\phi_1, \dots, \phi_{10})$. Specifically, we used a Dirichlet prior with a small concentration parameter, as in Sec. 4.5, in order to eliminate 5 redundant components:

$$\begin{aligned} (\phi_1, \dots, \phi_{10}) &\sim \mathrm{Dirichlet}(\alpha_1, ..., \alpha_{10}), \\ \theta_i^{(N)} &:= \phi_i * 500, \quad (i = 1, \dots, 10) \end{aligned}$$

where $\alpha_1 = \cdots = \alpha_{10} = 0.01$ denote the concentration parameters. For each of $\theta_1^{(T)}, \dots, \theta_{10}^{(T)}$, we defined a broad uniform prior $\theta_i^{(T)} \sim \mathrm{Unif}(0, 480)$.

*Table 4.* Results for the pedestrian simulator in Sec. 4.6

| Algorithm | $\theta^{(N)}$ error | $\theta^{(T)}$ error | data error | cputime |
|---|---|---|---|---|
| KR-ABC | 61.58(74.42) | 70.93(102.08) | 0.008(0.009) | 2233.45(97.54) |
| KR-ABC (less) | 82.46(75.05) | 134.00(161.85) | 0.014(0.014) | 1875.32(147.16) |
| K2-ABC | 298.94(120.71) | 308.95(109.43) | 0.10(0.10) | 1547.32(56.31) |
| K-ABC | 354.72(145.76) | 389.52(140.91) | 0.12(0.09) | 1773.74(84.91) |
| SMC-ABC (mean) | 271.51(104.64) | 363.12(91.28) | 0.09(0.07) | 2017.89(110.02) |
| SMC-ABC (MAP) | 255.15(139.33) | 348.43(104.74) | 0.09(0.1) | 2017.89(110.02) |
| ABC-DC | 273.93(136.14) | 327.48(98.12) | 0.09(0.14) | 1984.43(59.12) |
| BO | 194.57(65.83) | 291.73(105.33) | 0.04(0.06) | 37541.23(3047.46) |
| MSM | 453.58(89.43) | 510.04(55.10) | 0.24(0.17) | 1869.83(49.51) |

From the true model, we simulated 4200 time steps of pedestrian flow as observed data. We made $5 \times 5 = 25$ grids in a map of Ginza and computed a histogram of the corresponding 25 bins for each time step. Thus, observed data was made up of 4200 vectors in $\mathbb{R}^{25}$. In the same way, each method generated a total of 4200 vectors, and each iterative method made use of 200 vectors in each iteration, running 21 iterations in total. For SMC-ABC, we set the parameter $\alpha \in (0, 1)$ to be 0.2, as in the previous experiment.

We ran each method 20 times, and the resulting averages and standard deviations for errors are summarized in Table 4, where "$\theta^{(N)}$ error" and "$\theta^{(T)}$ error" denote the errors of the corresponding estimated parameters, as measured in terms of Euclidean distance. These errors were computed in the same way as in Sec. 4.5 (e.g., the estimated parameters were sorted according to the magnitudes of the mixing coefficients). Results show that our method performed the best, confirming its effectiveness. In the Supplementary Materials, we also report the point estimates made using the proposed method, showing that the true parameters were estimated reasonably accurately.

## 5. Summary and future work

We have proposed kernel recursive ABC for point estimation with intractable likelihood and have empirically investigated the effectiveness of this approach. While we have also provided theoretical analysis to a certain extent, there remain important theoretical topics, as discussed in Sec. 3.1, that we wish to reserve for future research.

### Acknowledgements

We thank the anonymous reviewers as well as Itaru Nishioka, Itsuki Noda, Takashi Washio, Shinji Ito, Wittawat Jitkrittum, and Marie Oshima for their helpful comments and support. We also thank Shuhei Mano for providing his code. MK has been supported by the European Research Council (StG Project PANAMA). KF has been supported by JSPS KAKENHI 26280009.

[5] https://github.com/crest-cassia/CrowdWalk

## References

Bach, F., Lacoste-Julien, S., and Obozinski, G. (2012). On the equivalence between herding and conditional gradient algorithms. In *Proceedings of the 29th International Conference on Machine Learning (ICML2012)*, pages 1359–1366.

Beaumont, M. A., Zhang, W., and Balding, D. J. (2002). Approximate Bayesian computation in population genetics. *Genetics*, **162**(4), 2025–2035.

Brochu, E., Cora, V. M., and De Freitas, N. (2010). A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*.

Chen, Y., Welling, M., and Smola, A. (2010). Super-samples from kernel herding. In *Proceedings of the Twenty-Sixth Conference on Uncertainty in Artificial Intelligence*, pages 109–116. AUAI Press.

Dean, T. A., Singh, S. S., Jasra, A., and Peters, G. W. (2014). Parameter estimation for hidden Markov models with intractable likelihoods. *Scandinavian Journal of Statistics*, **41**(4), 970–987.

Del Moral, P., Doucet, A., and Jasra, A. (2012). An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and Computing*, **22**(5), 1009–1020.

Dick, J., Kuo, F. Y., and Sloan, I. H. (2013). High dimensional numerical integration - The Quasi-Monte Carlo way. *Acta Numerica*, **22**(133-288).

Doucet, A., Godsill, S. J., and Robert, C. P. (2002). Marginal maximum a posteriori estimation using Markov chain Monte Carlo. *Statistics and Computing*, **12**(1), 77–84.

Fukumizu, K., Gretton, A., Sun, X., and Schölkopf, B. (2008). Kernel measures of conditional dependence. In *Advances in neural information processing systems*, pages 489–496.

Fukumizu, K., Song, L., and Gretton, A. (2013). Kernel Bayes' rule: Bayesian inference with positive definite kernels. *The Journal of Machine Learning Research*, **14**(1), 3753–3783.

Fukunaga, K. and Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on information theory*, **21**(1), 32–40.

Gourieroux, C., Monfort, A., and Renault, E. (1993). Indirect inference. *Journal of applied econometrics*, **8**(S1).

Gretton, A., Borgwardt, K. M., Rasch, M., Schölkopf, B., and Smola, A. J. (2007). A kernel method for the two-sample-problem. In *Advances in neural information processing systems*, pages 513–520.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. (2012). A kernel two-sample test. *Journal of Machine Learning Research*, **13**(Mar), 723–773.

Grünewälder, S., Lever, G., Baldassarre, L., Patterson, S., Gretton, A., and Pontil, M. (2012). Conditional mean embeddings as regressors. In *Proceedings of the 29th International Conference on Machine Learning (ICML2012)*, pages 1823–1830.

Gutmann, M. U. and Corander, J. (2016). Bayesian optimization for likelihood-free inference of simulator-based statistical models. *Journal of Machine Learning Research*, **17**(125), 1–47.

Kanagawa, M., Sriperumbudur, B. K., and Fukumizu, K. (2016a). Convergence guarantees for kernel-based quadrature rules in misspecified settings. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 3288–3296. Curran Associates, Inc.

Kanagawa, M., Nishiyama, Y., Gretton, A., and Fukumizu, K. (2016b). Filtering with state-observation examples via kernel Monte Carlo filter. *Neural Computation*, **28**(2), 382–444.

Lacoste-Julien, S., Lindsten, F., and Bach, F. (2015). Sequential kernel herding: Frank-Wolfe optimization for particle filtering. In G. Lebanon and S. V. N. Vishwanathan, editors, *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, volume 38 of *Proceedings of Machine Learning Research*, pages 544–552. PMLR.

Lele, S. R., Nadeem, K., , and Schmuland, B. (2010). Estimability and likelihood inference for generalized linear mixed models using data cloning. *Journal of the American Statistical Association*, **105**(492), 1617–1625.

Marjoram, P., Molitor, J., Plagnol, V., and Tavaré, S. (2003). Markov chain Monte Carlo without likelihood. *Proceedings of the National Academy of Sciences*, **100**(26), 15324–15328.

McFadden, D. (1989). A method of simulated moments for estimation of discrete response models without numerical integration. *Econometrica: Journal of the Econometric Society*, pages 995–1026.

Meeds, E. and Welling, M. (2014). GPS-ABC: Gaussian process surrogate approximate Bayesian computation. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, pages 593–602. AUAI Press.

Mitrovic, J., Sejdinovic, D., and Teh, Y.-W. (2016). DR-ABC: approximate Bayesian computation with kernel-based distribution regression. In *International Conference on Machine Learning*, pages 1482–1491.

Muandet, K., Fukumizu, K., Sriperumbudur, B. K., and Schölkopf, B. (2017). Kernel mean embedding of distributions : A review and beyond. *Foundations and Trends in Machine Learning*, **10**(1–2), 1–141.

Nakagome, S., Fukumizu, K., and Mano, S. (2013). Kernel approximate Bayesian computation in population genetic inferences. *Statistical applications in genetics and molecular biology*, **12**(6), 667–678.

Nolan, J. P. (2013). Multivariate elliptically contoured stable distributions: theory and estimation. *Computational Statistics*, **28**(5), 2067–2089.

Park, M., Jitkrittum, W., and Sejdinovic, D. (2016). K2-ABC: Approximate Bayesian computation with kernel embeddings. In A. Gretton and C. C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 398–407, Cadiz, Spain. PMLR.

Picchini, U. and Anderson, R. (2017). Approximate maximum likelihood estimation using data-cloning abc. *Computational Statistics & Data Analysis*, **105**, 166–183.

Pritchard, J. K., Seielstad, M. T., Perez-Lezaun, A., and Feldman, M. W. (1999). Population growth of human y chromosomes: a study of y chromosome microsatellites. *Molecular biology and evolution*, **16**(12), 1791–1798.

Rubio, F. J., Johansen, A. M., *et al.* (2013). A simple approach to maximum intractable likelihood estimation. *Electronic Journal of Statistics*, **7**, 1632–1654.

Smola, A., Gretton, A., Song, L., and Schölkopf, B. (2007). A Hilbert space embedding for distributions. In *Proceedings of the International Conference on Algorithmic Learning Theory*, volume 4754, pages 13–31. Springer.

Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical Bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959.

Song, L., Huang, J., Smola, A., and Fukumizu, K. (2009). Hilbert space embeddings of conditional distributions with applications to dynamical systems. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 961–968. ACM.

Song, L., Fukumizu, K., and Gretton, A. (2013). Kernel embeddings of conditional distributions: A unified kernel framework for nonparametric inference in graphical models. *IEEE Signal Processing Magazine*, **30**(4), 98–111.

Sriperumbudur, B. K., Gretton, A., Fukumizu, K., Schölkopf, B., and Lanckriet, G. R. (2010). Hilbert space embeddings and metrics on probability measures. *Jounal of Machine Learning Research*, **11**, 1517–1561.

Székely, G. J. and Rizzo, M. L. (2013). Energy statistics: A class of statistics based on distances. *Journal of statistical planning and inference*, **143**(8), 1249–1272.

Takeuchi, I., Le, Q. V., Sears, T. D., and Smola, A. J. (2006). Nonparametric quantile estimation. *Journal of Machine Learning Research*, **7**(Jul), 1231–1264.

Tavaré, S., Balding, D. J., Griffiths, R. C., and Donnelly, P. (1997). Inferring coalescence times from DNA sequence data. *Genetics*, **145**(2), 505–518.

Toni, T., Welch, D., Strelkowa, N., Ipsen, A., and Stumpf, M. P. (2009). Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface*, **6**(31), 187–202.

Wood, S. N. (2010). Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, **466**(7310), 1120–4.

Yamashita, T., Soeda, S., and Noda, I. (2010). Assistance of evacuation planning with high-speed network model-based pedestrian simulator. In *Proceedings of Fifth International Conference on Pedestrian and Evacuation Dynamics (PED 2010)*, page 58. PED 2010.

Yamazaki, K. and Kaji, D. (2013). Comparing two Bayes methods based on the free energy functions in Bernoulli mixtures. *Neural Networks*, **44**, 36–43.

Yıldırım, S., Singh, S. S., Dean, T., and Jasra, A. (2015). Parameter estimation in hidden Markov models with intractable likelihoods using sequential Monte Carlo. *Journal of Computational and Graphical Statistics*, **24**(3), 846–865.