

Gaussian process modeling in approximate Bayesian computation to estimate horizontal gene transfer in bacteria

Marko Järvenpää¹, Michael U. Gutmann², Aki Vehtari¹, and Pekka Marttinen¹

¹Helsinki Institute for Information Technology HIIT, Department of Computer Science, Aalto University

²School of Informatics, University of Edinburgh

June 22, 2017

Abstract

Approximate Bayesian computation (ABC) can be used for model fitting when the likelihood function is intractable but simulating from the model is feasible. However, even a single evaluation of a complex model may take several hours, limiting the number of model evaluations available. Modelling the discrepancy between the simulated and observed data using a Gaussian process (GP) can be used to reduce the number of model evaluations required by ABC, but the sensitivity of this approach to a specific GP formulation has not yet been thoroughly investigated. We begin with a comprehensive empirical evaluation of using GPs in ABC, including various transformations of the discrepancies and two novel GP formulations. Our results indicate the choice of GP may significantly affect the accuracy of the estimated posterior distribution. Selection of an appropriate GP model is thus important. We formulate expected utility to measure the accuracy of classifying discrepancies below or above the ABC threshold, and show that it can be used to automate the GP model selection step. Finally, based on the understanding gained with toy examples, we fit a population genetic model for bacteria, providing insight into horizontal gene transfer events within the population and from external origins.

1 Introduction

Estimating parameters of a statistical model often requires evaluating the likelihood function. For complex models, such as those arising in population genetics, deriving or evaluating the likelihood in a reasonable computation time may be impossible. On the other hand, generating data from the model may be relatively straightforward. Approximate Bayesian Computation (ABC) [Beaumont et al., 2002, Hartig et al., 2011, Marin et al., 2012, Turner and Van Zandt, 2012, Lintusaari et al., 2016] is an inference framework for such models. It is based on generating data from the simulation model for various parameter values and comparing the simulated data with the observed data using some discrepancy measure. The simplest ABC algorithm is the rejection sampler, which, at each step, randomly simulates a parameter from the prior distribution, runs the simulation model with this parameter, and finally accepts the parameter if the discrepancy between the simulated and observed data is smaller than some threshold. These steps are repeated until a sufficient number of samples from the approximate posterior have been collected.

To speed up ABC inference, several sampling-based algorithms have been proposed [Marjoram et al., 2003, Sisson et al., 2007, Beaumont et al., 2009, Toni et al., 2009, Moral et al., 2012, Lenormand et al., 2013]. An alternative to sampling that has received much attention in recent years is to construct an explicit approximation to the likelihood function, and use this as a proxy for the exact likelihood in e.g. MCMC samplers. In the synthetic likelihood method this is done by modelling the summary statistics with a multivariate Gaussian [Wood, 2010, Price et al., 2016], see also Fan et al. [2013], Papamakarios and Murray

[2016] for some other approaches. Nonparametric approximations have also been considered [Blum, 2010, Turner and Sederberg, 2014], and connections to other approaches are discussed by Drovandi et al. [2015b], Gutmann and Corander [2016]. Gaussian processes [Rasmussen and Williams, 2006] (GPs) can naturally encode assumptions about the smoothness of the likelihood. They have been used by Drovandi et al. [2015a] to accelerate pseudo-marginal MCMC methods, and by Wilkinson [2014], Kandasamy et al. [2015] to model the likelihood function. An alternative is to model individual summaries with a GP [Meeds and Welling, 2014, Jabot et al., 2014].

Typically hundreds of thousands of model simulations are needed for ABC inference, but here we focus on the challenging case where less than a thousand evaluations are available due to computational constraints. We adopt the approach of Gutmann and Corander [2016] who modelled the discrepancy between observed and simulated data with a GP, which has the advantage that computing the ABC posterior estimate can be done even with relatively few model evaluations. Also, modelling the scalar-valued discrepancy allows one to use Bayesian optimisation [Brochu et al., 2010, Shahriari et al., 2015] to effectively select evaluation locations [Gutmann and Corander, 2016]. The ABC posterior is proportional to the product of the prior and the probability that the simulated discrepancy falls below the threshold, and this quantity can be computed analytically from the fitted GP. However, a potential issue in using a GP to model the discrepancies is that, in practice, the GP modelling assumptions may not hold exactly [Gutmann and Corander, 2016]. The discrepancy is often positive (e.g. a weighted Euclidean distance), non-Gaussian, and its variance may vary over the parameter space, causing additional approximation error of unknown magnitude. In this article we study this in detail. To focus on the GP modelling aspect, we assume the that region of non-negligible posterior probability is known in advance, but acknowledge that detecting the region is a topic of ongoing research on its own [Wilkinson, 2014, Kandasamy et al., 2015, Gutmann and Corander, 2016, Järvenpää et al., 2017].

The impact of GP model assumptions on the resulting ABC posterior is demonstrated with a realistic example in Figure 1, where different GP formulations are used to model the discrepancy in the area with non-negligible posterior probability. The model here describes horizontal gene transfer between bacterial genomes, published recently by Marttinen et al. [2015]. The discrepancies were obtained by fixing other parameters to their point estimates, and generating realisations of the discrepancy with varying values for a parameter that describes the frequency of horizontal gene transfer between bacteria. A thorough analysis of the model is presented in Section 3.3. For now please note that the input-dependent noise model [Goldberg et al., 1997, Tolvanen et al., 2014] is able to take into account the heteroscedastic variance of the discrepancy and, consequently, seems to result in an accurate approximation to the posterior (the true posterior is here unavailable). On the other hand, with the standard GP regression the fit is poor, and the resulting posterior distribution appears too wide.

Our paper makes the following contributions:

- Motivated by the preliminary investigation with the population genetics model above, we assess the impact of the GP formulation on ABC inference for multiple benchmark models.
- We propose two generalisations of previously presented GP-ABC approaches: first, we allow heteroscedastic noise in the GP; second, we use a classifier GP to directly model the probability of the discrepancy being below the ABC threshold.
- We propose a new utility function to automate GP model choice for ABC. The utility function favours models that achieve higher accuracy in classifying discrepancies below or above the ABC threshold.
- As a practical application, we derive an accurate posterior distribution for the population genetic model for gene transfer in bacteria, allowing us to make inferences about the relationship between gene deletions and introductions, and between gene transfers from within the population and from external origins.

This paper is organised as follows. In Section 2 we briefly review general ABC methods and introduce different GP models for ABC. We also discuss GP model selection in ABC. In Section 3 we present findings

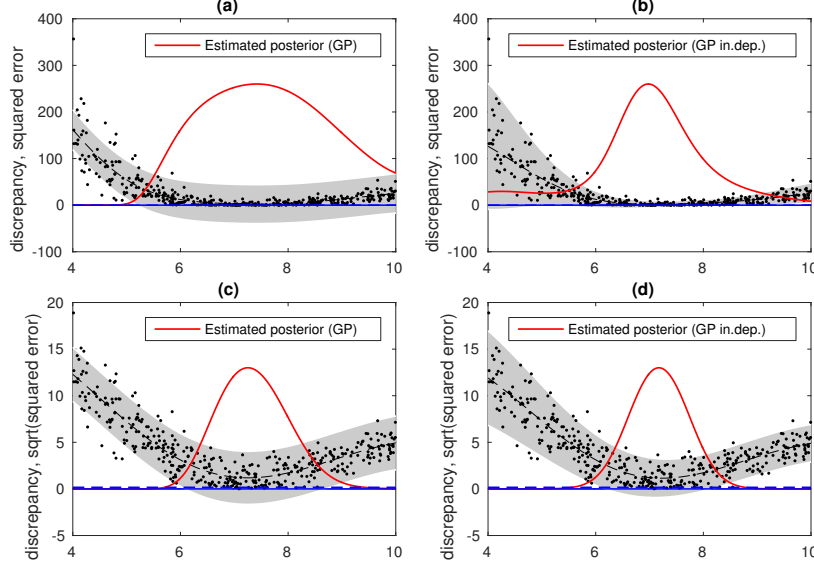


Figure 1: The GP surrogate used to model the simulated discrepancies affects the accuracy of the resulting ABC posterior estimate. x -axis is the simulation model parameter and y -axis the value of the (transformed) discrepancy. Black dots are the simulated realisations of discrepancies, and the grey area is the 95% predictive interval, representing stochastic variation in the simulation. The red line shows the corresponding posterior approximation which is computed as a lower tail probability from the discrepancy model. (a) The standard GP results in overestimated variance of the discrepancy, yielding a poor approximation to the posterior. Input-dependent GP model in (b) or discrepancy transformation in (c) result in better approximations. The best fit is here obtained when using both the transformation and the input-dependent GP model in (d), as even after the square-root transformation in (c) the variance of the discrepancy is not constant.

from multiple example problems to illustrate the impact of GP assumptions and model selection in ABC, and finally present the results for the bacterial genomics model. Section 4 contains discussion and in Section 5 we conclude with recommendations on handling GP surrogates in ABC inference.

2 Background and methods

2.1 ABC

We assume that we have observed data $\mathbf{y} \in \mathbb{R}^d$ from a simulation model whose likelihood function can be written as $p(\mathbf{y} | \boldsymbol{\theta})$, where the unknown parameters to be estimated are $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^p$, and the prior density is $p(\boldsymbol{\theta})$. The posterior distribution can then be computed from the Bayes' theorem

$$p(\boldsymbol{\theta} | \mathbf{y}) = \frac{p(\boldsymbol{\theta})p(\mathbf{y} | \boldsymbol{\theta})}{\int p(\boldsymbol{\theta}')p(\mathbf{y} | \boldsymbol{\theta}') d\boldsymbol{\theta}'} \propto p(\boldsymbol{\theta})p(\mathbf{y} | \boldsymbol{\theta}). \quad (1)$$

When either the analytic form of the likelihood function $p(\mathbf{y} | \boldsymbol{\theta})$ is unavailable or its value cannot be evaluated in a reasonable time, the standard alternative is to use approximate Bayesian computation (ABC). The ABC targets the approximate posterior

$$p_{\text{ABC}}(\boldsymbol{\theta} | \mathbf{y}) \propto p(\boldsymbol{\theta}) \int \mathbf{1}_{\Delta(\mathbf{y}, \mathbf{x}) \leq \varepsilon} p(\mathbf{x} | \boldsymbol{\theta}) d\mathbf{x}, \quad (2)$$

where $\mathbf{x} \in \mathbb{R}^d$ denotes pseudo-data generated by the simulation model with parameter $\boldsymbol{\theta}$. The pseudo-data \mathbf{x} are compared to the observed data \mathbf{y} and $\Delta(\cdot, \cdot)$ is a discrepancy function between the two data sets.

In practice, the threshold ε represents a tradeoff between estimation accuracy and efficiency; small values result in more accurate estimates but require more computation. The discrepancy is often formed using some summary statistics such that if s is a mapping from the data space \mathbb{R}^d to a lower dimensional space of the summary statistics, then $\Delta(\mathbf{y}, \mathbf{x}) = \|s(\mathbf{y}) - s(\mathbf{x})\|$. Choosing informative summaries and combining them in a reasonable way affect the resulting approximate posterior [Marin et al., 2012, Fearnhead and Prangle, 2012] but we do not consider this problem here.

Given N samples from the simulation model with a chosen parameter $\boldsymbol{\theta}$, so that $\mathbf{x}_{\boldsymbol{\theta}}^{(i)} \sim p(\mathbf{x} | \boldsymbol{\theta})$, $i = 1, \dots, N$, the ABC posterior at $\boldsymbol{\theta}$ can be estimated using

$$p_{\text{ABC}}(\boldsymbol{\theta} | \mathbf{y}) \propto p(\boldsymbol{\theta}) \sum_{i=1}^N \mathbb{1}_{\Delta(\mathbf{y}, \mathbf{x}_{\boldsymbol{\theta}}^{(i)}) \leq \varepsilon}. \quad (3)$$

Alternatively, one can use ABC rejection sampling to sample from the ABC posterior, with the following steps: 1. Draw $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta})$, 2. Generate $\mathbf{x}^{(i)} \sim p(\mathbf{x} | \boldsymbol{\theta}^{(i)})$ from the simulation model, 3. Accept $\boldsymbol{\theta}^{(i)}$ if $\Delta(\mathbf{y}, \mathbf{x}_{\boldsymbol{\theta}^{(i)}}) \leq \varepsilon$. The accepted values $\{\boldsymbol{\theta}^{(i)}\}$ are samples from the approximate posterior distribution. For further background on ABC, we refer the reader to the recent review by Lintusaari et al. [2016].

2.2 BOLFI method

To speedup inference, Gutmann and Corander [2016] proposed to model the discrepancy $\Delta_{\boldsymbol{\theta}} = \Delta(\mathbf{y}, \mathbf{x}_{\boldsymbol{\theta}})$ between the observed data \mathbf{y} and the simulated data $\mathbf{x}_{\boldsymbol{\theta}}$ as a function of $\boldsymbol{\theta}$. At each step of their algorithm, the current training data i.e. the discrepancy-parameter pairs $D_t = \{(\Delta^{(i)}, \boldsymbol{\theta}^{(i)})\}_{i=1}^t$, are used to train the discrepancy model, which is then used to intelligently select the next parameter value $\boldsymbol{\theta}^{(t+1)}$ to run the computationally costly simulation model, and thus to obtain updated training data D_{t+1} . The simulations can be adaptively focused to areas yielding small discrepancy values (*exploitation*), while allowing some *exploration* of new areas with potentially small values.

At each step, the fitted surrogate model is used to compute an estimated ABC posterior. As opposed to Eq. 3, the estimated posterior for each $\boldsymbol{\theta}$ can be obtained as $p(\boldsymbol{\theta}) \mathbb{P}(\Delta_{\boldsymbol{\theta}} \leq \varepsilon)$, where the probability is computed using the statistical model (i.e. the fitted GP). For any continuous and strictly increasing function g , it holds that $\mathbb{P}(\Delta_{\boldsymbol{\theta}} \leq \varepsilon) = \mathbb{P}(g(\Delta_{\boldsymbol{\theta}}) \leq \varepsilon')$, where $\varepsilon' = g(\varepsilon)$. Thus one can also model $g(\Delta_{\boldsymbol{\theta}})$ instead to $\Delta_{\boldsymbol{\theta}}$, which facilitates straightforward transformations for the discrepancy (e.g. the logarithm) possibly making the discrepancy easier to model.

2.3 GP models for ABC

In this section we describe different GP formulations for modelling the (possibly transformed) discrepancy in the BOLFI approach. In addition to the standard GP model, we include two novel extensions (see below): the input-dependent GP and the classifier GP. We assume that the training data consists of discrepancy-parameter pairs $D_t = \{(\Delta^{(i)}, \boldsymbol{\theta}^{(i)})\}_{i=1}^t$ from the modal area of the posterior, and the aim is to model the discrepancy and the resulting posterior as accurately as possible using D_t .

In the **standard GP regression** one assumes that $\Delta_{\boldsymbol{\theta}} \sim \mathcal{N}(f(\boldsymbol{\theta}), \sigma^2)$ and $f(\boldsymbol{\theta}) \sim \mathcal{GP}(m(\boldsymbol{\theta}), k(\boldsymbol{\theta}, \boldsymbol{\theta}'))$ with a mean function $m : \boldsymbol{\Theta} \rightarrow \mathbb{R}$ and covariance function $k : \boldsymbol{\Theta} \times \boldsymbol{\Theta} \rightarrow \mathbb{R}$. We use $m(\boldsymbol{\theta}) = 0$, and the squared exponential covariance function $k(\boldsymbol{\theta}, \boldsymbol{\theta}') = \sigma_f^2 \exp(-\sum_{i=1}^p (\theta_i - \theta'_i)^2 / (2l_i^2))$ in our experiments. Given the hyperparameters $\boldsymbol{\phi} = (\sigma_f^2, l_1, \dots, l_p, \sigma^2)$ and training data D_t , the posterior predictive density for the latent function f at $\boldsymbol{\theta}$ follows a Gaussian density with mean and variance

$$\mu_t(\boldsymbol{\theta}) = \mathbf{k}_t(\boldsymbol{\theta})^T \mathbf{K}_t^{-1}(\boldsymbol{\theta}) \Delta^{(1:t)}, \quad v_t(\boldsymbol{\theta}) = k(\boldsymbol{\theta}, \boldsymbol{\theta}) - \mathbf{k}_t(\boldsymbol{\theta})^T \mathbf{K}_t^{-1}(\boldsymbol{\theta}) \mathbf{k}_t(\boldsymbol{\theta}), \quad (4)$$

respectively. Above we have denoted $\mathbf{k}_t(\boldsymbol{\theta}) = (k(\boldsymbol{\theta}, \boldsymbol{\theta}^{(1)}), \dots, k(\boldsymbol{\theta}, \boldsymbol{\theta}^{(t)}))^T$, $[\mathbf{K}_t(\boldsymbol{\theta})]_{ij} = k(\boldsymbol{\theta}^{(i)}, \boldsymbol{\theta}^{(j)}) + \sigma^2 \mathbb{1}_{i=j}$ for $i, j = 1, \dots, t$ and $\Delta^{(1:t)} = (\Delta^{(1)}, \dots, \Delta^{(t)})^T$. The hyperparameters $\boldsymbol{\phi}$ are estimated by maximising the

marginal likelihood, for details, see [Rasmussen and Williams \[2006\]](#). A model-based estimate of the likelihood at θ can be obtained from the fitted GP as

$$\mathbb{P}(\Delta_\theta \leq \varepsilon) = \Phi((\varepsilon - \mu_t(\theta))/\sqrt{v_t(\theta) + \sigma^2}), \quad (5)$$

where ε is the threshold and Φ is the cumulative distribution function of the standard Gaussian distribution. An estimate of the posterior density is obtained by multiplying the estimated likelihood with the prior $p(\theta)$.

Next we describe the **input-dependent GP model** [[Goldberg et al., 1997](#), [Tolvanen et al., 2014](#)]. In the standard GP model the noise variance σ^2 representing the stochasticity in the discrepancy due to simulation is assumed constant. We relax this by assuming $\Delta_\theta \sim \mathcal{N}(f(\theta), \sigma^2 \exp(g(\theta)))$, $f(\theta) \sim \mathcal{GP}(m(\theta), k(\theta, \theta'))$ and $g(\theta) \sim \mathcal{GP}(m_n(\theta), k_n(\theta, \theta'))$. That is, also the variance of the discrepancy is modelled with a GP allowing it to change smoothly as a function of the parameter θ . Since the variance must be positive, its logarithm is modelled with the GP. As before we set $m(\theta) = 0$, and also $m_n(\theta) = 0$, implying that a priori the average variance is close to σ^2 . We use the squared exponential covariance functions $k(\theta, \theta') = \sigma_f^2 \exp(-\sum_{i=1}^p (\theta_i - \theta'_i)^2 / (2l_{f_i}^2))$ and $k_n(\theta, \theta') = \sigma_g^2 \exp(-\sum_{i=1}^p (\theta_i - \theta'_i)^2 / (2l_{g_i}^2))$. There are $2p + 2$ hyperparameters to be estimated: p lengthscales parameters, l_{g_i} , l_{f_i} , and one signal variance parameter for each covariance function, σ_f^2 , σ_g^2 . The value of σ^2 is fixed to make the covariance hyperparameters identifiable. Laplace approximation is used for model fitting. We also experimented with the expectation propagation approximation by [Tolvanen et al. \[2014\]](#), but this came with additional cost and results were qualitatively similar. Eq. 5 can still be used to estimate the likelihood, by replacing the point estimate of σ^2 with an estimate of $\sigma^2 \exp(g(\theta))$.

The GP models above are used for modelling the ABC discrepancy between observed and simulated data. However, for computing the approximate posterior, it is sufficient to know the probability that the discrepancy is below the threshold ε . Motivated by this, we propose a method, **classifier GP**, which models the lower tail probability directly as a function of the parameter θ , using binary GP classification. We interpret the observations $z_i = 2\mathbb{1}_{\Delta^{(i)} \leq \varepsilon} - 1$ as class labels $+1$ and -1 such that $p(z_i | f(\theta_i)) = \lambda^{-1}(z_i f(\theta_i))$, where λ is either the logit or probit link function and $f(\theta) \sim \mathcal{GP}(m(\theta), k(\theta, \theta'))$. Hence, this corresponds to an assumption that the discriminative function is smooth, but does not impose additional assumptions about the distribution of the discrepancy. For each parameter value θ the model thus specifies the probability of the discrepancy being classified as $+1$, i.e., to be below the threshold. The likelihood estimate is thus obtained directly. Unlike with other GP models, we add an additional constant to the prior mean function $m(\theta)$ to take into account the fact that the lower tail probabilities are generally very small. Without this, the discriminative function tended to become nonzero near the parameter bounds, inducing posterior mass near the boundaries and, consequently, poor approximations. We use the squared exponential covariance function for the latent function f as for the standard GP method, and Laplace approximation model fitting, see [Rasmussen and Williams \[2006\]](#) for details.

2.4 GP model selection

Since the distribution of the discrepancy depends on the characteristics of the simulation model and the chosen discrepancy, some GP models will fit the training data better than others. Consequently, we propose two utility functions for comparing GP models and different transformations of discrepancy, with the aim of choosing the GP formulation that yields the most accurate estimate of the posterior. See e.g. [Bernardo and Smith \[2001\]](#), [Vehtari and Ojanen \[2012\]](#) for a thorough discussion on using expected utility for model selection.

As the first criterion, we consider the expected log predictive density for a new discrepancy value $\Delta^{(t+h)}$ evaluated at some future evaluation point $\theta^{(t+h)}$ for $h = 1, 2, \dots$. Here the utility of a single observation $\Delta^{(t+h)}$ is defined by

$$u_h = \log p(\Delta^{(t+h)} | \theta^{(t+h)}, D^{(1:t)}, M), \quad (6)$$

where $D^{(1:t)} = \{(\Delta^{(i)}, \theta^{(i)})\}_{i=1}^t$ denotes the training data gathered thus far and M denotes the model. The different transformations of the discrepancy are taken into account by considering the effect of the

transformation $\Delta' = g(\Delta)$. The expected utility estimate is obtained by averaging over all the possible realisations of the future data yielding $\bar{u} = \mathbb{E}_h(u_h)$. This utility measures how well the GP predicts the distribution of the discrepancies, which is used for computing the posterior estimate of the simulation model. As we do not know the distribution of the discrepancy-parameter data $(\Delta^{(t+h)}, \boldsymbol{\theta}^{(t+h)})$, we approximate the expected utilities using the data $D^{(1:t)}$ [Vehtari and Lampinen, 2002]. K -fold cross-validation (CV) leads to the following estimate for expected log predictive density

$$\bar{u}_{\text{CV}} = \frac{1}{t} \sum_{i=1}^t \log p(\Delta^{(i)} | \boldsymbol{\theta}^{(i)}, D^{(1:t) \setminus s(i)}, M), \quad (7)$$

where the data are split into K (almost) equally sized groups and $s(i)$ denotes the indexes of the group to which the i th data point belongs. In practice, we use $K = 10$. In the sequel, we refer to this as the **mlpd utility**, which stands for the mean of the log-predictive density.

A downside of the mlpd utility is that it does not acknowledge the final purpose of the selected GP model, i.e., to approximate the posterior distribution. It may thus give high scores to GP models which broadly model the discrepancy accurately, whereas the focus should be on how well the smallest discrepancies are modelled, as those affect the posterior approximation most. Motivated by this, we frame the problem as a classification task which then leads to a new utility function tailored for ABC inference. The utility for a single observation $\Delta^{(t+h)}$ is defined by

$$u_h^c = \mathbb{1}_{\Delta^{(t+h)} \leq \varepsilon} \log(\mathbb{P}(\Delta^{(t+h)} \leq \varepsilon | M)) + \mathbb{1}_{\Delta^{(t+h)} > \varepsilon} \log(\mathbb{P}(\Delta^{(t+h)} > \varepsilon | M)), \quad (8)$$

where $\mathbb{P}(\Delta^{(t+h)} \leq \varepsilon | M)$ is the probability that a new realisation of the discrepancy $\Delta^{(t+h)}$ at a test point $\boldsymbol{\theta}^{(t+h)}$ is smaller than the threshold ε according to model M (conditioning on $\boldsymbol{\theta}^{(t+h)}$ and $D^{(1:t)}$ is omitted to simplify notation). This utility penalises realisations of the discrepancy that are under the threshold when, according to the model, this should happen only with a very small probability, or vice versa. An additional advantage of this utility is that it is invariant to a transformation of the discrepancy if the threshold ε is transformed accordingly, and also it can be used to compare the classifier GP to other models, as it only requires the probability that the discrepancy is below the threshold. Again, we use the K -fold CV with $K = 10$ to approximate the expected utility, so that

$$\bar{u}_{\text{CV}}^c = \frac{1}{t} \sum_{i=1}^t (\mathbb{1}_{\Delta^{(i)} \leq \varepsilon} \log(\mathbb{P}(\Delta^{(i)} \leq \varepsilon | D^{(1:t) \setminus s(i)}, M)) + \mathbb{1}_{\Delta^{(i)} > \varepsilon} \log(\mathbb{P}(\Delta^{(i)} > \varepsilon | D^{(1:t) \setminus s(i)}, M))). \quad (9)$$

We call this the **classifier utility** from now on.

3 Results

3.1 Toy examples

We consider several toy examples to study the approximation error for different GP models and transformations of the discrepancy. A summary of the test problems is given in Table 1. Although simple, these examples highlight potential challenges in modelling the discrepancy that we expect carry over to many realistic problems of potentially higher dimensionality. The quality of the results is assessed by computing the total variation distance (TV) between the estimated and the corresponding true posterior i.e. $\text{TV}(p_{\text{true}}, p_{\text{approx}}) = 1/2 \int_{\Theta} |p_{\text{true}}(\boldsymbol{\theta}) - p_{\text{approx}}(\boldsymbol{\theta})| d\boldsymbol{\theta}$. Values for this integral are computed numerically. Kullback-Leibler divergence (KL), defined as $\text{KL}(p_{\text{true}} || p_{\text{approx}}) = \int_{\Theta} p_{\text{true}}(\boldsymbol{\theta}) \log(p_{\text{true}}(\boldsymbol{\theta})/p_{\text{approx}}(\boldsymbol{\theta})) d\boldsymbol{\theta}$ is used as an alternative criterion and is also computed numerically. GPstuff 4.6 [Vanhatalo et al., 2013] is used for fitting the GP models.

We consider two transformations of the squared error (*se*) discrepancy shown in Table 1, namely, the log and the square-root transformations (*log* and *sqrt*). Although other transformations can be used, these already demonstrate the main findings. One could also transform the individual summaries before combining

Test problem and model	prior	n	discrepancy Δ_θ	true θ
Gaussian 1, $\mathcal{N}(\theta, 1)$	$\mathcal{U}([-0.5, 3])$	10	$(\bar{y} - \bar{y}(\theta))^2$	1
Bimodal, $\mathcal{N}(\theta^2, 2)$	$\mathcal{U}([-2.5, 2.5])$	5	$(\bar{y} - \bar{y}(\theta))^2$	± 1
Gaussian 2, $\mathcal{N}(0, \theta)$	$\mathcal{U}([0, 5])$	10	$(\sigma_y^2 - \sigma_{y(\theta)}^2)^2$	1
Poisson, $\text{Poi}(\theta)$	$\mathcal{U}([0, 5])$	10	$(\bar{y} - \bar{y}(\theta))^2$	2
GM 1, $\mathcal{GM}(0.7, \theta, \theta + 5, 1, 2)$	$\mathcal{U}([-10, 5])$	1	$(y_1 - y_1(\theta))^2$	1
GM 2, $\mathcal{GM}(0.7, \theta, \theta, 3, 0.25)$	$\mathcal{U}([-6, 6])$	1	$(y_1 - y_1(\theta))^2$	1
Uniform, $\mathcal{U}([0, \theta])$	$\mathcal{U}([0, 5])$	5	$(\max\{y_i\} - \max\{y_i(\theta)\})^2$	2
2D Gaussian 1, $\mathcal{N}(\theta, \Sigma)$	$\mathcal{U}([1.5, 4] \times [1.5, 4])$	10	$(\bar{\mathbf{y}} - \bar{\mathbf{y}}(\theta))^T \Sigma^{-1} (\bar{\mathbf{y}} - \bar{\mathbf{y}}(\theta))$	$[2.5, 2.5]^T$
2D Gaussian 2, $\mathcal{N}(\theta_1, \theta_2)$	$\mathcal{U}([2, 4.5] \times [0.5, 5])$	25	$(\bar{y} - \bar{y}(\theta))^2 + (\sigma_y^2 - \sigma_{y(\theta)}^2)^2$	$[3, 2]^T$
Lotka-Volterra, see the text	$\mathcal{U}([0.25, 1.25] \times [0.5, 1.5])$	8	see the text	$[1, 1]^T$

Table 1: Description of the test problems. Above, \bar{y} denotes the sample mean of $\{y_i\}_{i=1}^n$ and, similarly, σ_y^2 is the sample variance. The data points $y_i(\theta)$ are independent and identically distributed draws from the simulation model with parameter θ . Also, $\mathcal{GM}(\alpha, \mu_1, \mu_2, \sigma_1^2, \sigma_2^2) = \alpha \mathcal{N}(\mu_1, \sigma_1^2) + (1 - \alpha) \mathcal{N}(\mu_2, \sigma_2^2)$. For the 2D Gaussian we use a fixed covariance matrix Σ with unit variances and correlation 0.5.

them to a discrepancy function, but we do not consider this approach here. We set the ABC threshold ε customarily as the 0.05th quantile of the sampled discrepancies and use the same threshold for all GP-ABC methods, the baseline ABC rejection sampler, and for the "true" ABC posterior computed using ABC rejection sampling with extensive simulations. Thus the difference in results is only caused by the choice of GP model and discrepancy transformation. We also repeat the experiments with some other choices of the threshold and using the true posterior density (which is available analytically for the test problems) as the baseline. These results are presented in the appendices A.2 and A.3. For the basic ABC rejection sampler, we use kernel density estimation as a post-processing step to approximate the posterior curve from the accepted samples, thereby imposing basic smoothness assumptions of the posterior. We use the logit link function for GP classifier method in our experiments.

We use uniform priors for the parameters of the simulation models over a range covering the modal area of the true posterior. We also repeat the experiments with a much wider support, although this seems less relevant in practice when the goal is to obtain an accurate posterior estimate where the majority of mass is located, and hence focus simulations there. We note that other priors and adaptive schemes for choosing the training data are also possible [Gutmann and Corander, 2016]. To get an estimate of the variability due to a stochastic simulation model, we repeat each experiment 100 times. The full summary of the results is gathered in Tables 2 and 3, and below we analyse in detail some of the key findings.

Example 3.1 (Gaussian 1). As the first example, representing many key findings, we consider a simple Gaussian model with an unknown mean and known variance, see 'Gaussian 1' in Table 1. This model is simple enough to be analysed analytically. Consider the discrepancies $\Delta_\theta = (\bar{y} - \bar{x}_\theta)^2$ and $\Delta'_\theta = |\bar{y} - \bar{x}_\theta|$. Using basic properties of the expectation and the Gaussian distribution, we obtain $\mathbb{E}(\Delta_\theta) = (\theta - \bar{y})^2 + \sigma^2/n$ and $\text{var}(\Delta_\theta) = 2\sigma^2(2(\theta - \bar{y})^2 + 1)/n$. Similarly $\text{var}(\Delta'_\theta) \approx \sigma^2/n$, which holds accurately for large $|\theta|$. We see that the variance of the discrepancy Δ_θ grows quadratically as a function of the parameter θ . On the other hand, with Δ'_θ the variance is approximately constant.

The main observations of this example are illustrated in Figure 2. In (a-c) a prior over a wide range is used and in (d-f) training data are gathered within a narrower region around the mode. Comparing (a) and (b) shows that the input-dependent GP model yields a much better approximation than the standard GP. In (a) the poor GP fit causes also a poor approximation to the posterior, which cannot be corrected by increasing the number of simulations. Furthermore, different transformations change the behaviour of the discrepancy. The square-root-transformation in (c), which makes the variance of the discrepancy approximately constant, improves the standard GP considerably. In (d-f) three different GP models are fitted near the posterior mode,

and we see that focusing the simulations to the central region improves the performance of all methods.

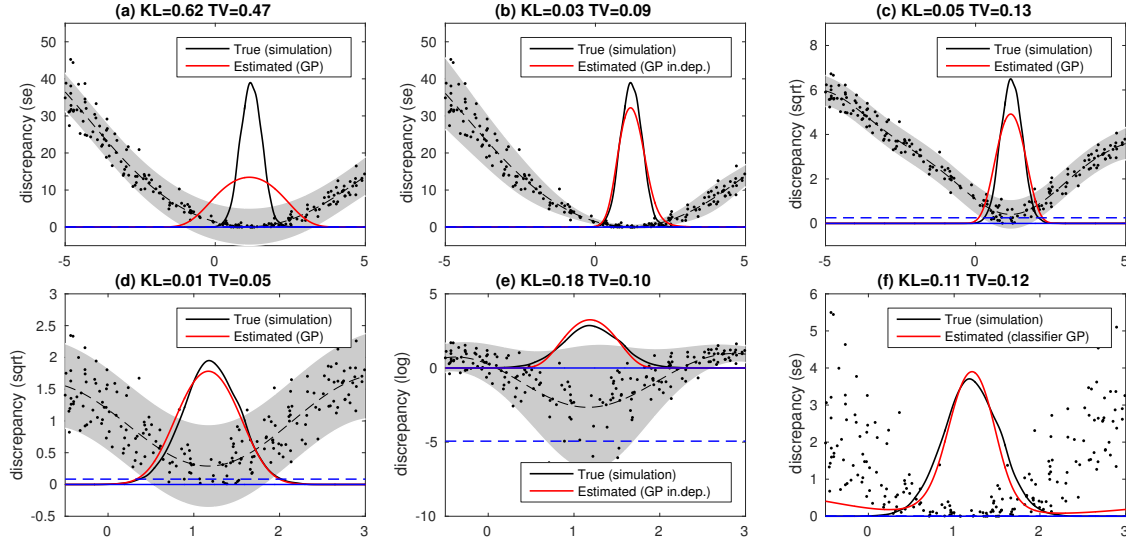


Figure 2: Results for the 'Gaussian 1' model. The grey area is the 95% probability interval, blue dashed line is the threshold and the black dots represent realisations of the discrepancy. The abbreviations 'se', 'log' and 'sqrt' refer to squared, log transformed and square-root transformed discrepancies, respectively. In (a) the GP is fitted to discrepancy realisations on a wide interval resulting in a poor approximation. Better approximations are obtained by the input-dependent GP model (b) or transforming the discrepancy (c). In (d) the fitting is done on the area of significant posterior mass resulting in the best fit in terms of both TV and KL, even if the variance of the discrepancy is still clearly overestimated in the modal region. In (e) the posterior uncertainty is slightly underestimated due to the skewness of the log-transformed discrepancy. The classifier GP in (f) slightly overestimates the tails of the posterior.

Example 3.2 (Poisson). We estimate the parameter of the Poisson distribution which demonstrates the benefit of GP modelling compared to the ABC rejection sampling. Figure 3 shows typical results. Here the data have discrete values but the discrepancy is approximately Gaussian, and the variance of the discrepancy grows as a function of the parameter. The input-dependent model does not improve the results visibly, even if the fit to the discrepancy data is evidently better. The best approximations are obtained when the square-root transformation is used as in (a-b) since then the discrepancy is approximately Gaussian, although its variance is not constant. The ABC rejection sampler in (c) does not work well due to the small number of accepted samples as compared to the GP-based methods.

Example 3.3 (GM 1). The third example demonstrates that both the standard and input-dependent GP models may fail to capture a bimodal shape of the posterior. Here also the discrepancy distribution is bimodal conditional on specific parameter values. A particular realisation is shown in Figure 4. The GP and input-dependent GP yield slightly different approximations, neither of which captures the bimodality. However, fixing the lengthscale to a small value allows to capture the bimodal shape but with the cost of making the overall shape of the estimated posterior wiggly (not shown). The ABC rejection sampler works better despite the limited training set size of 200. This observation does not hold for all bimodal posteriors, though. To demonstrate this, we designed another example where the posterior is bimodal, the 'Bimodal' in Table 1. In contrast with the Gaussian mixture model above, the distribution of the discrepancy is close to a Gaussian for any parameter. This type of discrepancy can be modelled well, and consequently, the bimodal shape of the posterior can be learnt accurately.

Example 3.4 (Lotka-Volterra). We consider the Lotka-Volterra model used by Toni et al. [2009] to compare ABC methods. The model describes the evolution of prey and predator populations, defined by differential

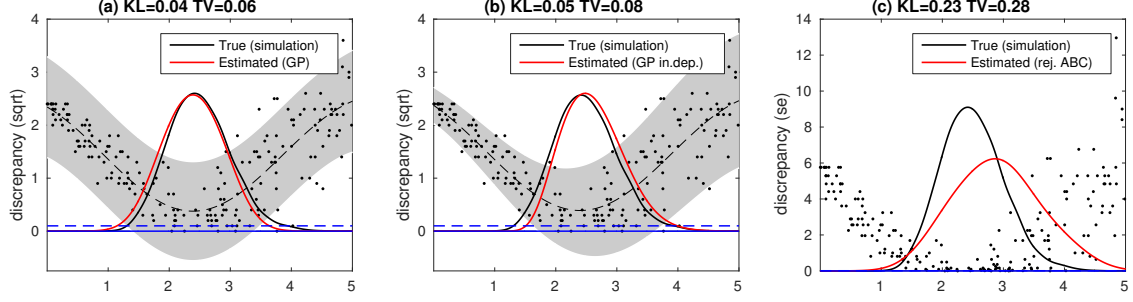


Figure 3: Results for the 'Poisson' example, demonstrating the benefits of the GP modelling. The input-dependent GP model (b) fits the discrepancy data better than the standard GP (a). Despite this difference, the posterior approximations are about equally good. The ABC rejection sampler (c) yields a less accurate approximation with only the 200 training points available.

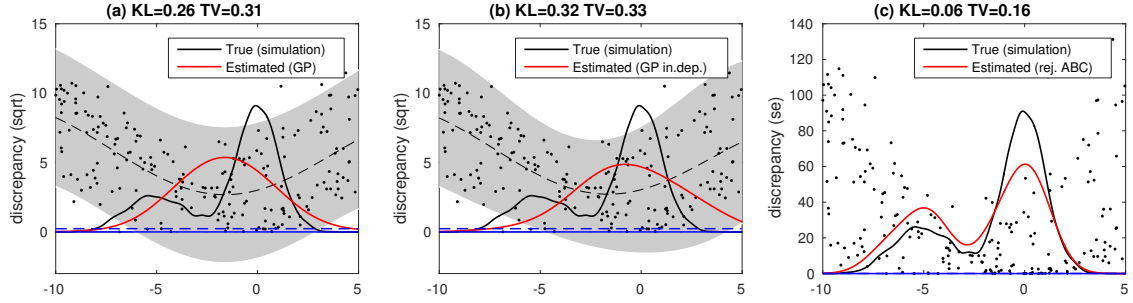


Figure 4: Neither the standard (a) nor the input-dependent GP model (b) learn the shape of the posterior in the bimodal 'GM 1' example. Notably, not only the posterior, but also the discrepancy distribution, given a particular parameter value, is bimodal, which is the explanation of this behaviour. 200 points were generated from the model, but similar results were obtained with a larger set of simulations, and with other transformations. On the other hand, the ABC rejection sampler (c) uncovers the bimodal shape.

equations

$$\frac{dx_1}{dt} = \theta_1 x_1 - x_1 x_2, \quad \frac{dx_2}{dt} = \theta_2 x_1 x_2 - x_2, \quad (10)$$

where $x_1 = x_1(t)$ and $x_2 = x_2(t)$ describe the prey and predator species at time t , respectively. Their initial values are set to $x_1(0) = 0.5$ and $x_2(0) = 1.0$. Vector $\theta = (\theta_1, \theta_2)$ is the parameter to be estimated. The 8 measurements for (x_1, x_2) are corrupted by additive independent and identically distributed Gaussian noise $\mathcal{N}(0, 0.5^2)$. We consider a discrepancy

$$\Delta_{\theta} = \sum_{i=1}^8 \sum_{j=1}^2 (x_j(t_i) - \hat{x}_j(t_i, \theta))^2, \quad (11)$$

where $x_j(t_i)$ are the noisy measurements at time t_i and $\hat{x}_j(t_i, \theta)$ the corresponding predictions with parameter θ . The prior and the true value of the parameter vector are shown in Table 1.

The results are shown in Figure 5, and we see that the GP formulation has only a moderate impact. However, the classifier GP and the ABC rejection sampler perform worse than the GP-based methods, as seen also in Table 3. The estimates of the ABC rejection sampler also vary more between the different simulated training data.

	n=50			n=100			n=200			n=400			n=600		
	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt
Gaussian 1:															
GP	0.17	0.09	0.07	0.20	0.10	0.05	0.21	0.11	0.04	0.20	0.17	0.03	0.20	0.18	0.03
GP in.dep.	0.19	0.14	0.09	0.18	0.14	0.06	0.18	0.13	0.05	0.19	0.14	0.04	0.21	0.15	0.04
classifier GP	0.40	0.40	0.40	0.33	0.33	0.33	0.19	0.19	0.19	0.10	0.10	0.10	0.09	0.09	0.09
rej. ABC	0.31	0.31	0.31	0.26	0.26	0.26	0.18	0.18	0.18	0.12	0.12	0.12	0.11	0.11	0.11
Bimodal:															
GP	0.20	0.47	0.16	0.20	0.26	0.12	0.21	0.18	0.10	0.21	0.16	0.08	0.20	0.17	0.07
GP in.dep.	0.19	0.58	0.18	0.17	0.39	0.14	0.20	0.28	0.11	0.20	0.25	0.09	0.20	0.23	0.08
classifier GP	0.39	0.39	0.39	0.35	0.35	0.35	0.24	0.24	0.24	0.17	0.17	0.17	0.14	0.14	0.14
rej. ABC	0.45	0.45	0.45	0.39	0.39	0.39	0.26	0.26	0.26	0.20	0.20	0.20	0.16	0.16	0.16
Gaussian 2:															
GP	0.32	0.36	0.27	0.32	0.24	0.26	0.33	0.20	0.25	0.33	0.21	0.24	0.32	0.22	0.23
GP in.dep.	0.48	0.37	0.30	0.45	0.33	0.29	0.44	0.29	0.27	0.42	0.25	0.26	0.41	0.25	0.25
classifier GP	0.35	0.35	0.35	0.35	0.35	0.35	0.26	0.26	0.26	0.17	0.17	0.17	0.15	0.15	0.15
rej. ABC	0.29	0.29	0.29	0.29	0.29	0.29	0.22	0.22	0.22	0.18	0.18	0.18	0.16	0.16	0.16
GM 1:															
GP	0.32	0.29	0.31	0.31	0.28	0.30	0.31	0.28	0.30	0.30	0.23	0.29	0.29	0.19	0.28
GP in.dep.	0.42	0.35	0.32	0.40	0.37	0.31	0.39	0.34	0.30	0.38	0.25	0.29	0.37	0.24	0.28
classifier GP	0.34	0.34	0.34	0.36	0.36	0.36	0.27	0.27	0.27	0.18	0.18	0.18	0.14	0.14	0.14
rej. ABC	0.35	0.35	0.35	0.32	0.32	0.32	0.24	0.24	0.24	0.19	0.19	0.19	0.14	0.14	0.14
GM 2:															
GP	0.19	0.14	0.13	0.18	0.12	0.12	0.19	0.12	0.11	0.19	0.11	0.11	0.20	0.11	0.11
GP in.dep.	0.26	0.19	0.14	0.24	0.18	0.12	0.21	0.16	0.11	0.22	0.16	0.10	0.24	0.15	0.10
classifier GP	0.37	0.37	0.37	0.33	0.33	0.33	0.21	0.21	0.21	0.14	0.14	0.14	0.12	0.12	0.12
rej. ABC	0.33	0.33	0.33	0.29	0.29	0.29	0.20	0.20	0.20	0.16	0.16	0.16	0.14	0.14	0.14
Uniform															
GP	0.26	0.22	0.15	0.26	0.24	0.15	0.27	0.22	0.15	0.26	0.23	0.15	0.26	0.23	0.15
GP in.dep.	0.26	0.22	0.16	0.22	0.21	0.13	0.19	0.23	0.12	0.17	0.23	0.11	0.15	0.23	0.12
classifier GP	0.43	0.43	0.43	0.34	0.34	0.34	0.23	0.23	0.13	0.14	0.14	0.14	0.11	0.11	0.11
rej. ABC	0.33	0.33	0.33	0.31	0.31	0.31	0.23	0.23	0.23	0.19	0.19	0.19	0.17	0.17	0.17
Poisson:															
GP	0.19	0.12	0.09	0.18	0.10	0.07	0.18	0.08	0.06	0.20	0.11	0.06	0.20	0.12	0.06
GP in.dep.	0.21	0.23	0.13	0.21	0.19	0.10	0.24	0.18	0.08	0.23	0.15	0.07	0.24	0.16	0.07
classifier GP	0.33	0.33	0.33	0.28	0.28	0.28	0.14	0.14	0.14	0.09	0.09	0.09	0.09	0.09	0.09
rej. ABC	0.26	0.26	0.26	0.23	0.23	0.23	0.16	0.16	0.16	0.11	0.11	0.11	0.10	0.10	0.10

Table 2: Results for the 1D toy examples. The quality of the approximation was measured using the TV distance between the estimated and the true ABC posterior densities. The smallest TV values are bolded. Value n is the number of model simulations and 'se', 'log' and 'sqrt' refer to squared, log transformed and square-root transformed discrepancies, respectively.

	n=100			n=200			n=400			n=600			n=800		
	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt
2D Gaussian 1:															
GP	0.24	0.15	0.12	0.23	0.13	0.09	0.22	0.12	0.07	0.22	0.12	0.07	0.22	0.12	0.07
GP in.dep.	0.20	0.20	0.14	0.19	0.15	0.10	0.17	0.12	0.08	0.18	0.12	0.07	0.19	0.11	0.07
classifier GP	0.62	0.62	0.62	0.63	0.63	0.63	0.24	0.24	0.24	0.15	0.15	0.15	0.12	0.12	0.12
rej. ABC	0.35	0.35	0.35	0.27	0.27	0.27	0.22	0.22	0.22	0.19	0.19	0.19	0.17	0.17	0.17
2D Gaussian 2:															
GP	0.53	0.32	0.45	0.51	0.26	0.43	0.51	0.22	0.40	0.50	0.21	0.39	0.50	0.20	0.39
GP in.dep.	0.64	0.27	0.26	0.61	0.20	0.24	0.50	0.17	0.22	0.49	0.15	0.22	0.50	0.13	0.21
classifier GP	0.63	0.63	0.63	0.63	0.63	0.63	0.36	0.36	0.36	0.21	0.21	0.21	0.17	0.17	0.17
rej. ABC	0.34	0.34	0.34	0.29	0.29	0.29	0.25	0.25	0.25	0.20	0.20	0.20	0.19	0.19	0.19
Lotka-Volterra:															
GP	0.22	0.19	0.20	0.18	0.15	0.16	0.16	0.13	0.15	0.15	0.12	0.13	0.15	0.12	0.12
GP in.dep.	0.31	0.23	0.23	0.21	0.19	0.19	0.18	0.16	0.15	0.16	0.15	0.14	0.15	0.14	0.12
classifier GP	0.51	0.51	0.51	0.51	0.51	0.51	0.50	0.50	0.50	0.20	0.20	0.20	0.17	0.17	0.17
rej. ABC	0.33	0.33	0.33	0.26	0.26	0.26	0.22	0.22	0.22	0.20	0.20	0.20	0.18	0.18	0.18

Table 3: Results for the 2D toy examples. See the caption for Table 2 for details.

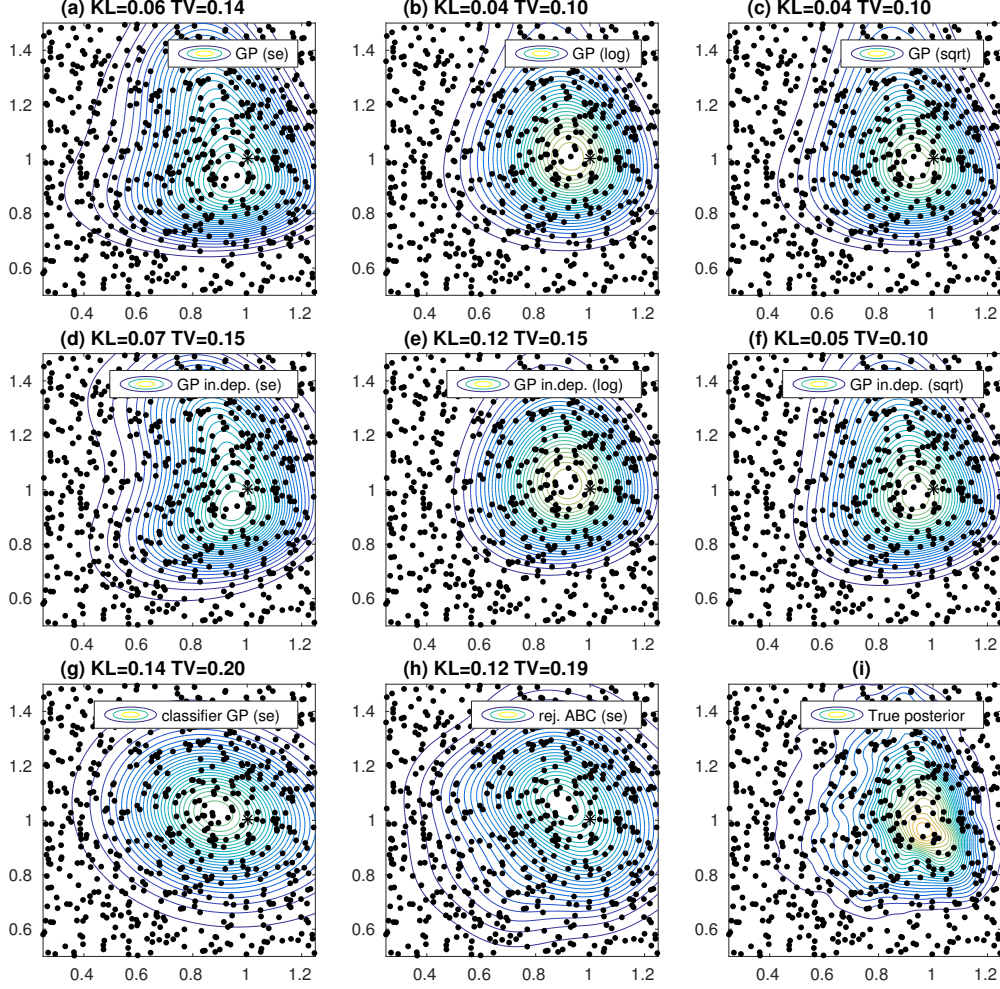


Figure 5: Posterior estimates for the Lotka-Volterra model. The black dots represent points where the simulation was run. The parameter θ_1 is on the x-axis and θ_2 on the y-axis. The difference between the standard and input-dependent GP formulations is minor, but both of them outperform the classifier GP and the ABC rejection sampler.

As a general observation from Tables 2 and 3, we conclude that whenever the discrepancy is close to a Gaussian, or if the number of evaluations is very small and only a few discrepancy values are below the threshold ε , the GP-based approaches yield better posterior approximations than the ABC rejection sampler or the classifier GP method. However, if the Gaussian assumptions are violated, as in Example 3.3, the rejection sampler and the classifier GP are more accurate. Increasing the number of simulations does not help as it does not solve the model misspecification. Interestingly, as few as 50 model evaluations in 1D (200 in 2D) result in almost as accurate results as 400 evaluations in 1D (600 in 2D). On the other hand, the accuracies of the ABC rejection sampler and classifier GP clearly improve as the number of evaluation points is increased. Additional evaluations also improve the stability of GP estimation and, hence, decrease the variance in the results.

The classifier GP performs generally similarly or slightly better than the ABC rejection sampler. However, with a small number of evaluations the error of the classifier GP is relatively large, but as the number of evaluations increases, the accuracy increases rapidly reaching and finally clearly outperforming the ABC rejection sampler. However, decreasing the threshold to the 0.01th quantile leads to conservative results

since the number of realisations of the discrepancy below the threshold becomes very small as shown in appendix. Typically the classifier GP tends to overestimate the probability in the posterior tail area despite our attempts to change this behaviour as described in Section 2.3.

Overall, the square-root transformation seems to work best while log-transformation is also useful in some cases. However, with small threshold values, such as the 0.01th quantile of the realised discrepancies, modelling the log-transformed discrepancy with a GP tends to cause too narrow posterior distributions in some scenarios, see Figure 2(e). This happens if many discrepancies are close to zero, in which case the log transformation results in a strongly skewed distribution. This may not be an issue in practice, since with a complex model simulating data such that the discrepancy becomes very small is unlikely (or impossible if the model is misspecified), even with the optimal parameter value. Also, modelling non-negative discrepancies with GP regression does not appear to cause large additional posterior approximation error in practice, see e.g. Figures 2 and 3. In some of the test cases, the input-dependent GP model worked best but a similar effect was often achieved also by modelling a suitably transformed discrepancy.

3.2 Model selection results

In Section 2.4, we formulated two utility functions to guide the selection of the GP model: the expected log predictive density (mlpd utility) and the expected log predictive probability of attaining a discrepancy that falls below the threshold (classifier utility). Next we illustrate the performance of these methods in practice. We consider the same toy problems as in Section 3.1, and we exclude the classifier GP from the comparisons related to the mlpd utility.

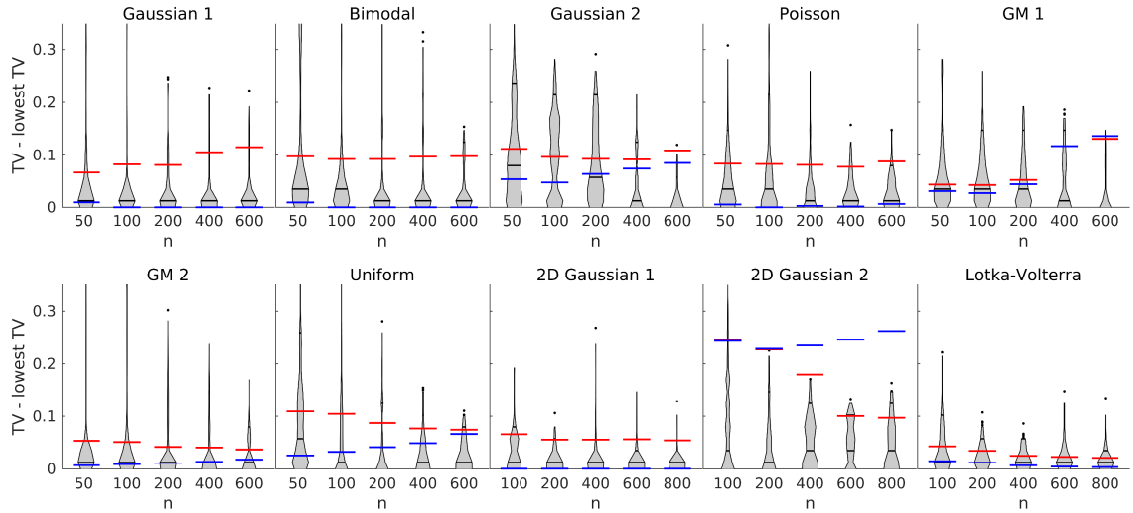


Figure 6: Results of the GP model selection using the classifier utility. The value on the y-axis is the difference between the TV distance of the chosen GP (corresponding to the largest utility) and the smallest TV distance observed (corresponding to the most accurate result obtained). Therefore, the smaller the value is, the closer the selected model is to the optimal model. The violin plot shows the results over 100 simulated training data sets. The x-axis shows the number of model simulations n . The blue line represents the median results if the standard GP with the square-root transformation is always chosen. Another baseline shown with red is obtained by randomly selecting the GP model formulation.

The results in Figure 6 and Figure 8 in appendix A.1 demonstrate the performance of the mlpd and classifier utilities, when used to select a GP model to estimate the posterior. We see that both methods work reasonably well across all cases, although 'Gaussian 2' and '2D Gaussian 1' toy problems seem more difficult than the rest. Also, as expected, as more simulations become available, the model selection improves in most scenarios, such that the highest utilities better identify the GP formulations resulting in the most

accurate posterior approximations. For some individual simulations a GP model resulting in a poor posterior approximation has the highest utility. This happens mainly with a small number of simulations and the classifier utility, because then the number of cases below the threshold is very small, and, consequently, the utility itself has a high variance. These cases are seen as peaks in the violin plots.

Comparison of Figure 6 and Figure 8 shows that the overall performance difference between the two proposed utilities is relatively small. However, in the case of ‘Gaussian 2’ and ‘GM 1’ examples, the mlpd utility performs systematically worse than the classifier utility. In these cases even with 600 evaluations, the mlpd utility tends to propose suboptimal GP models. Further, the classifier utility can be used to compare basically any set of models that predict the amount of posterior mass under the threshold, making it more applicable than the mlpd utility as explained in Section 2.4. On the other hand, the performance of the classifier utility criterion is more dependent on the value of the threshold. When the threshold is decreased so that only a few discrepancies fall below the threshold, the method will not work anymore, contrary to the mlpd utility.

3.3 Horizontal gene transfer between bacterial genomes

The emerging field of bacterial genomics involves analysis of thousands of bacterial genomes, to understand the variability in bacteria as well as to answer questions of practical importance, such as the spread of antibiotic resistance [Croucher et al., 2011, Chewapreecha et al., 2014]. One interesting observation is the extent to which members of the same bacterial species can differ in genome content, i.e., different strains of the same species can have different sets of genes, and only a minority of the genes is observed in all strains [Touchon et al., 2009]. Furthermore, bacteria can exchange genes with one another in a process called horizontal gene transfer (HGT) [Thomas and Nielsen, 2005].

Here we consider a previously published population genomic model that describes the variation in genome content [Marttinen et al., 2015]. Point estimates of the parameters have previously been published for this model, but we are interested in estimating the full posterior, when the model is fitted to a published collection of 616 genomes from *Streptococcus pneumoniae* [Croucher et al., 2013]. Briefly, the model consists of a forward-simulation of a population of bacterial strains for many generations. At each generation, the next generation is simulated by selecting strains randomly from the current generation. In addition, the genome content of the descendants may be modified by three operations, the rates of which correspond to the three parameters of the model: the gene deletion rate (*del*), novel gene introduction rate (*nov*), and the rate of HGT where the gene presence-absence status of the donor strain is copied to the recipient strain (*hgt*).

To estimate the model parameters, we consider the discrepancy

$$\Delta_{\theta} = w_1 \text{KL}(\theta) + w_2 (c_{\text{real}} - c_{\text{simu}}(\theta))^2, \quad (12)$$

where $\text{KL}(\theta)$ is the Kullback-Leibler divergence between the observed and simulated gene frequency spectra, c_{real} is the so-called observed clonality score, and $c_{\text{simu}}(\theta)$ the corresponding simulated value, see Marttinen et al. [2015] for details. The weights w_1 and w_2 are used to transform the summaries approximately on the same scale, which is common in ABC literature. Marttinen et al. [2015] achieved the same effect by log-transforming the KL-divergence, but up to this difference, the discrepancy here is the same as the one used by Marttinen et al. [2015]. Also, because the discrepancy has been investigated before, we are able to construct *a priori* plausible ranges for the parameters $\Theta = [0.01, 0.15] \times [0.1, 0.35] \times [4, 10]$, and we use the uniform prior $p(\theta) = \mathcal{U}(\Theta)$. For the GP computations the *hgt* parameter is scaled so that the parameters are approximately on the same scale. We run the simulation model in parallel with 1,000 points generated from the prior. Most simulations require one to two hours on a single processor. We set the threshold to the 0.05th quantile of the simulated discrepancy values, but the 0.01th quantile led to similar conclusions. We model the discrepancy using the standard and input-dependent GP models and the same transformations as in the previous sections.

The estimated posterior marginals are shown in Figure 7 and additional visualisation is included to the appendix B. The largest classifier utility score corresponds to the input-dependent GP model with the log transformation (classifier utility = -0.101) but also the square-root transformation with input-dependent

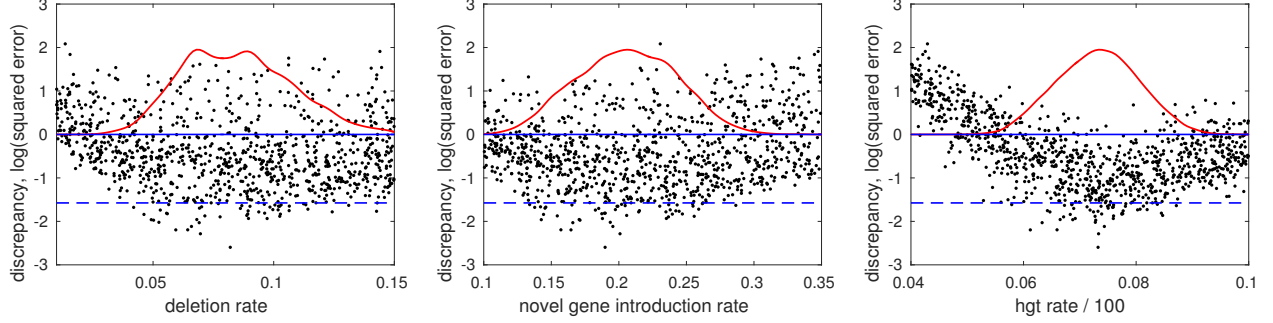


Figure 7: Marginal posterior densities for the three parameters of the genetics model. The discrepancy was log-transformed and the final model fitting was done by running the model 1000 times and using the input-dependent GP model. The black dots are the model simulations (projected to each coordinate axis), the dashed blue line is the threshold and solid blue line describes the zero line.

GP and the log transformation with standard GP yield visually similar approximations with utilities -0.102 and -0.106 , respectively. On the other hand, the squared discrepancy in Equation 12 as such is difficult to model, resulting in overestimated posterior uncertainty (see Figure 1). In general, the input-dependent GPs have higher utilities compared to the corresponding standard GPs for this simulation model. However, since we simulate only 1,000 training data points, we expect the posterior variance to still be slightly overestimated, as seen in many toy examples. We also note that the approximated posterior agrees well with the earlier reported point estimate $\theta = (0.066, 0.18, 7.4)$. In addition, we see a strong positive correlation ($\rho = 0.48$) between the *del* and *nov* parameters, which intuitively means that a high gene deletion rate can be compensated by a high rate of introducing novel genes into the population.

Finally, we derive posterior predictive distributions for two biologically interpretable quantities, i) the ratio between the number of all gene acquisitions vs. gene deletions (computed by considering all acquisitions and deletions, caused either by HGT within the population or a novel acquisition/deletion), and ii) the ratio of gene introductions to the population from outside the population (as novel genes) vs. from within the population (through HGT). The posterior predictive distributions are obtained by re-weighting the original simulations with importance sampling. The 95% credible interval for quantity i) is approximately $(1.17, 1.44)$ and for quantity ii) it is $(0.26, 0.52)$. Interestingly, we see that there are significantly more gene acquisitions than deletions, as with a high probability their ratio, the quantity i), is larger than one. Because in reality the genomes are not rapidly growing, this indicates some mechanism to counter the imbalance between acquisitions and deletions, for example selection against larger genomes in general, or alternatively that many new genes are individually selected against, see discussion by [Marttinen et al. \[2015\]](#). On the other hand, the ratio of gene acquisitions from outside vs. from within the population, the quantity ii), is approximately 0.4, which corresponds to the biological expectation that the majority of horizontal gene transfer events happens between closely related bacterial strains, see e.g. [Majewski \[2001\]](#), [Fraser et al. \[2007\]](#). To our knowledge, this has not been estimated before using simulation-based inference.

4 Discussion

We have thoroughly studied the use of GPs to enhance ABC inference, but, nevertheless, many choices could not be systematically investigated. We only considered the squared exponential covariance function, but expect the conclusions to hold also with other common options, as in [Jabot et al. \[2014\]](#). We also used a zero mean function unlike [Wilkinson \[2014\]](#), [Gutmann and Corander \[2016\]](#), who assumed that the discrepancy goes to infinity far from the minimum, and thus included quadratic terms to the mean function. Our choice allows for estimating posterior distributions of arbitrary shapes, at the cost of potentially overestimating

the tails of the distributions. The results also depend on the GP hyperparameters; we used relatively uninformative priors for them and estimated them by maximising the marginal likelihood. Integrating over the hyperparameters might improve the accuracy and stability, as in Snoek et al. [2012], and alleviate numerical problems, which we occasionally encountered especially with the input-dependent GP model. Difficult cases included heavy-tailed, bimodal, or skewed discrepancy distributions, and cases where the discrepancy was approximately constant in some region but grew rapidly elsewhere.

We further assumed the summary statistics and the discrepancy function given, but in practice they must be designed carefully. We also considered a fixed set of transformations of the discrepancy, but other choices, such as the warped GP regression [Snelson et al., 2004], could be used to derive additional transformations. Overall, the error caused by a poorly designed discrepancy may be larger than the approximation error caused by an unsuitable GP model. Nevertheless, we find it important to understand and try to minimise the approximation error introduced in the modelling phase. In order to focus on the GP modelling aspect, we further assumed that the region with non-negligible posterior probability was known approximately in advance. In practice this could be estimated by Bayesian optimisation with the standard GP model. An interesting future direction is to formally integrate adaptive model selection with acquisition of novel evaluation locations.

While our study is the first to compare different models for the discrepancies, other studies on modelling in ABC have been conducted before. For example, Blum and François [2010] modelled individual summary statistics for regression adjustment method in ABC and allowed heteroscedastic noise. Blum [2010] used different transformations of the summary statistics and investigated the selection of the corresponding regression adjustment method using cross-validation based criteria. The normality assumptions of the synthetic likelihood method [Wood, 2010] were examined by Price et al. [2016], and, similarly to us, inferences were found relatively robust to deviations from normality, except when the summaries had heavy tails or were bimodal. Jabot et al. [2014] compared different emulation methods for ABC, namely local regressions and GPs. However, unlike in this work, the authors modelled the summaries separately, as was done also by Meeds and Welling [2014].

We applied the techniques to a previously published population genetic model for horizontal gene transfer in bacteria [Marttinen et al., 2015]. In this realistic example, the input-dependent GP model with log-transformed discrepancies had the highest model selection utility, and was thus selected for presenting the results. This enabled us to derive the full posterior distribution for the parameters of the model. We estimated the number of gene acquisitions to be significantly higher than the number of gene deletions, suggesting some form of selection to prevent genomes from growing rapidly, to counterbalance this observation. We also estimated for the first time with simulation-based inference the ratio of gene transfers within the population considered, and those from external origins, and the results supported the empirical expectation that the majority of gene transfers happens between closely related strains. We note that multiple different models for bacterial evolution have been published, which differ in their purpose and assumptions [Fraser et al., 2007, Doroghazi and Buckley, 2011, Cohan and Perry, 2007, Shapiro et al., 2012, Ansari and Didelot, 2014, Niehus et al., 2015]. The methods considered here establish a sound basis for estimating parameters in these models and their possible future generalisations.

5 Conclusions

We considered the challenging task of ABC inference with a small number of model evaluations, and investigated the use of GPs to model the simulated discrepancies to fully use the scarce information available. Overall, we found this had a great potential to improve the accuracy of the posterior when the number of evaluations was limited. As anticipated by Gutmann and Corander [2016], we observed that the discrepancy distributions may in realistic situations deviate from standard GP assumptions, for example the variance may be heteroscedastic or the distribution skewed or multimodal. Motivated by this, we studied various GP formulations for modelling the discrepancies, or the probabilities of the discrepancies being below the ABC threshold. The input-dependent GP and classifier GP improved estimates in some scenarios and suitable transformations were also often helpful. Since no single approach was consistently the most accurate, we

designed a model selection criterion for choosing the GP formulation to use. We conclude by summarising our main findings:

- Squared discrepancies should be avoided due to their likely non-Gaussian distributions, and the dependence of the variance on the parameter, making them possibly difficult to model with a GP.
- Square-root and log-transformations often improve approximation quality. Similarly, the input-dependent GP improves the results if the variance of the discrepancy is not constant across the parameter space.
- In some cases none of the GP models may fit the data well, leading to poor posterior approximations. In these cases the classifier GP, the smoothed ABC rejection sampler, or some more general GP formulation not included here may be useful.
- Model selection tools can be used to select a GP model for ABC inference, and their accuracy improves along with the number of model simulations available.

Acknowledgement

This work was funded by the Academy of Finland (grants no. 286607 and 294015 to PM). We acknowledge the computational resources provided by the Aalto Science-IT project.

References

- M. A. Ansari and X. Didelot. Inference of the properties of the recombination process from whole bacterial genomes. *Genetics*, 196(1):253–265, 2014. [15](#)
- M. A. Beaumont, W. Zhang, and D. J. Balding. Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002. [1](#)
- M. A. Beaumont, J.-M. Cornuet, J.-M. Marin, and C. P. Robert. Adaptive approximate Bayesian computation. *Biometrika*, 96(4):983–990, 2009. [1](#)
- J. M. Bernardo and A. F. M. Smith. *Bayesian theory*. John Wiley & Sons, 2001. [5](#)
- M. G. B. Blum. Approximate Bayesian Computation: a nonparametric perspective. *Journal of American Statistical Association*, 105(491):1178–1187, 2010. [2](#), [15](#)
- M. G. B. Blum and O. François. Non-linear regression models for approximate bayesian computation. *Statistics and Computing*, 20(1):63–73, 2010. ISSN 1573-1375. doi: 10.1007/s11222-009-9116-0. URL <http://dx.doi.org/10.1007/s11222-009-9116-0>. [15](#)
- E. Brochu, V. M. Cora, and N. de Freitas. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. 2010. Available at <https://arxiv.org/abs/1012.2599>. [2](#)
- C. Chewapreecha, S. R. Harris, N. J. Croucher, C. Turner, P. Marttinen, L. Cheng, A. Pessia, D. M. Aanensen, A. E. Mather, A. J. Page, et al. Dense genomic sampling identifies highways of pneumococcal recombination. *Nature Genetics*, 46(3):305–309, 2014. [13](#)
- F. M. Cohan and E. B. Perry. A systematics for discovering the fundamental units of bacterial diversity. *Current Biology*, 17(10):R373–R386, 2007. [15](#)
- N. J. Croucher, S. R. Harris, C. Fraser, M. A. Quail, J. Burton, M. van der Linden, L. McGee, A. von Gottberg, J. H. Song, K. S. Ko, et al. Rapid pneumococcal evolution in response to clinical interventions. *Science*, 331(6016):430–434, 2011. [13](#)

- N. J. Croucher, J. A. Finkelstein, S. I. Pelton, P. K. Mitchell, G. M. Lee, J. Parkhill, S. D. Bentley, W. P. Hanage, and M. Lipsitch. Population genomics of post-vaccine changes in pneumococcal epidemiology. *Nature Genetics*, 45(6):656–663, 2013. 13
- J. R. Doroghazi and D. H. Buckley. A model for the effect of homologous recombination on microbial diversification. *Genome biology and evolution*, 3:1349–1356, 2011. 15
- C. C. Drovandi, M. T. Moores, and R. J. Boys. Accelerating pseudo-marginal MCMC using Gaussian processes. Available at <http://eprints.qut.edu.au/90973/>. Online. Accessed 19-10-2016, 2015a. 2
- C. C. Drovandi, A. N. Pettitt, and A. Lee. Bayesian indirect inference using a parametric auxiliary model. *Statistical Science*, 30(1):72–95, 2015b. 2
- Y. Fan, D. J. Nott, and S. A. Sisson. Approximate Bayesian computation via regression density estimation. *Stat*, 2(1):34–48, 2013. 1
- P. Fearnhead and D. Prangle. Constructing summary statistics for approximate Bayesian computation: Semi-automatic approximate Bayesian computation. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 74(3):419–474, 2012. 4
- C. Fraser, W. P. Hanage, and B. G. Spratt. Recombination and the nature of bacterial speciation. *Science*, 315(5811):476–480, 2007. 14, 15
- P. W. Goldberg, C. K. I. Williams, and C. M. Bishop. Regression with input-dependent noise: A Gaussian process treatment. *Advances in Neural Information Processing Systems*, 10:493–499, 1997. 2, 5
- M. U. Gutmann and J. Corander. Bayesian optimization for likelihood-free inference of simulator-based statistical models. *Journal of Machine Learning Research*, 17(125):1–47, 2016. 2, 4, 7, 14, 15
- F. Hartig, J. M. Calabrese, B. Reineking, T. Wiegand, and A. Huth. Statistical inference for stochastic simulation models—theory and application. *Ecology Letters*, 14(8):816–27, 2011. 1
- F. Jabot, G. Lagarrigues, B. Courbaud, and N. Dumoulin. A comparison of emulation methods for Approximate Bayesian Computation. Available at <http://arxiv.org/abs/1412.7560>, 2014. 2, 14, 15
- M. Järvenpää, M. Gutmann, A. Vehtari, and P. Marttinen. Efficient acquisition rules for model-based approximate Bayesian computation. Available at <https://arxiv.org/abs/1704.00520>, 2017. 2
- K. Kandasamy, J. Schneider, and B. Póczos. Bayesian active learning for posterior estimation. In *International Joint Conference on Artificial Intelligence*, pages 3605–3611, 2015. 2
- M. Lenormand, F. Jabot, and G. Deffuant. Adaptive approximate Bayesian computation for complex models. *Computational Statistics*, 28(6):2777–2796, 2013. 1
- J. Lintusaari, M. U. Gutmann, R. Dutta, S. Kaski, and J. Corander. Fundamentals and Recent Developments in Approximate Bayesian Computation. *Systematic biology*, 66(1):e66–e82, 2016. 1, 4
- J. Majewski. Sexual isolation in bacteria. *FEMS Microbiology Letters*, 199(2):161–169, 2001. 14
- J. M. Marin, P. Pudlo, C. P. Robert, and R. J. Ryder. Approximate Bayesian computational methods. *Statistics and Computing*, 22(6):1167–1180, 2012. 1, 4
- P. Marjoram, J. Molitor, V. Plagnol, and S. Tavaré. Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences of the United States of America*, 100(26):15324–8, 2003. 1
- P. Marttinen, M. U. Gutmann, N. J. Croucher, W. P. Hanage, and J. Corander. Recombination produces coherent bacterial species clusters in both core and accessory genomes. *Microbial Genomics*, 1(5), 2015. 2, 13, 14, 15

- E. Meeds and M. Welling. GPS-ABC: Gaussian Process Surrogate Approximate Bayesian Computation. In *Proceedings of the 30th Conference on Uncertainty in Artificial Intelligence*, 2014. 2, 15
- P. Moral, A. Doucet, and A. Jasra. An adaptive sequential Monte Carlo method for approximate Bayesian computation. *Statistics and Computing*, 22(5):1009–1020, 2012. 1
- R. Niehus, S. Mitri, A. G. Fletcher, and K. R. Foster. Migration and horizontal gene transfer divide microbial genomes into multiple niches. *Nature Communications*, 6, 2015. 15
- G. Papamakarios and I. Murray. Fast e-free inference of simulation models with Bayesian conditional density estimation. In *Advances in Neural Information Processing Systems 29*, 2016. 1
- L. F. Price, C. C. Drovandi, A. Lee, and D. J. Nott. Bayesian synthetic likelihood. Available at <http://eprints.qut.edu.au/92795/> Online. Accessed 19-10-2016, 2016. 1, 15
- C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006. 2, 5
- B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1), 2015. 2
- B. J. Shapiro, J. Friedman, O. X. Cordero, S. P. Preheim, S. C. Timberlake, G. Szabó, M. F. Polz, and E. J. Alm. Population genomics of early events in the ecological differentiation of bacteria. *Science*, 336(6077):48–51, 2012. 15
- S. A. Sisson, Y. Fan, and M. M. Tanaka. Sequential Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences of the United States of America*, 104(6):1760–5, 2007. 1
- E. Snelson, C. E. Rasmussen, and Z. Ghahramani. Warped Gaussian processes. In *Advances in Neural Information Processing Systems 16*, pages 337–344, 2004. 15
- J. Snoek, H. Larochelle, and R. P. Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems 25*, pages 1–9, 2012. 15
- C. M. Thomas and K. M. Nielsen. Mechanisms of, and barriers to, horizontal gene transfer between bacteria. *Nature Reviews Microbiology*, 3(9):711–721, 2005. 13
- V. Tolvanen, P. Jylänki, and A. Vehtari. Approximate inference for nonstationary heteroscedastic Gaussian process regression. In *2014 IEEE International Workshop on Machine Learning for Signal Processing*, pages 1–24, 2014. 2, 5
- T. Toni, D. Welch, N. Strelkowa, A. Ipsen, and M. P. H. Stumpf. Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society, Interface*, 6(31):187–202, 2009. 1, 8
- M. Touchon, C. Hoede, O. Tenaillon, V. Barbe, S. Baeriswyl, P. Bidet, E. Bingen, S. Bonacorsi, C. Bouchier, O. Bouvet, et al. Organised genome dynamics in the escherichia coli species results in highly diverse adaptive paths. *PLoS Genetics*, 5(1):e1000344, 2009. 13
- B. M. Turner and P. B. Sederberg. A generalized, likelihood-free method for posterior estimation. *Psychonomic Bulletin & Review*, 21(2):227–250, 2014. 2
- B. M. Turner and T. Van Zandt. A tutorial on approximate Bayesian computation. *Journal of Mathematical Psychology*, 56(2):69–85, 2012. 1
- J. Vanhatalo, J. Riihimäki, J. Hartikainen, P. Jylänki, V. Tolvanen, and A. Vehtari. GPstuff: Bayesian modeling with Gaussian processes. *Journal of Machine Learning Research*, 14:1175–1179, 2013. 6

- A. Vehtari and J. Lampinen. Bayesian model assessment and comparison using cross-validation predictive densities. *Neural computation*, 14(10):2439–2468, 2002. 6
- A. Vehtari and J. Ojanen. A survey of bayesian predictive methods for model assessment, selection and comparison. *Statistics Surveys*, 6:142–228, 2012. 5
- R. D. Wilkinson. Accelerating ABC methods using Gaussian processes. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, pages 1015–1023, 2014. 2, 14
- S. N. Wood. Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, 466:1102–1104, 2010. 1, 15

A Additional results

A.1 GP model selection for ABC

We present the experimental results for the mlpd utility in Figure 8. As discussed in the main text, overall the results look similar to those of the classifier utility, but with some test problems suboptimal GP models are systematically proposed.

Figures 9 and 10 show the GP model selection results when the baseline is the true posterior instead of the ABC posterior that was used in the main text and in Figure 8. Overall the results are similar but we observe some inconsistencies in the results of both utilities in the case of "2D Gaussian 1" and "Lotka-Volterra" test problems. This happens because the log-transformation produces too narrow posterior estimates compared to the corresponding true ABC posterior and thus poor approximations to it. However, the tendency to produce too narrow posterior estimates compensates the approximation error caused by the nonzero threshold and thus the log-transform produces the best estimates compared to the true posterior.

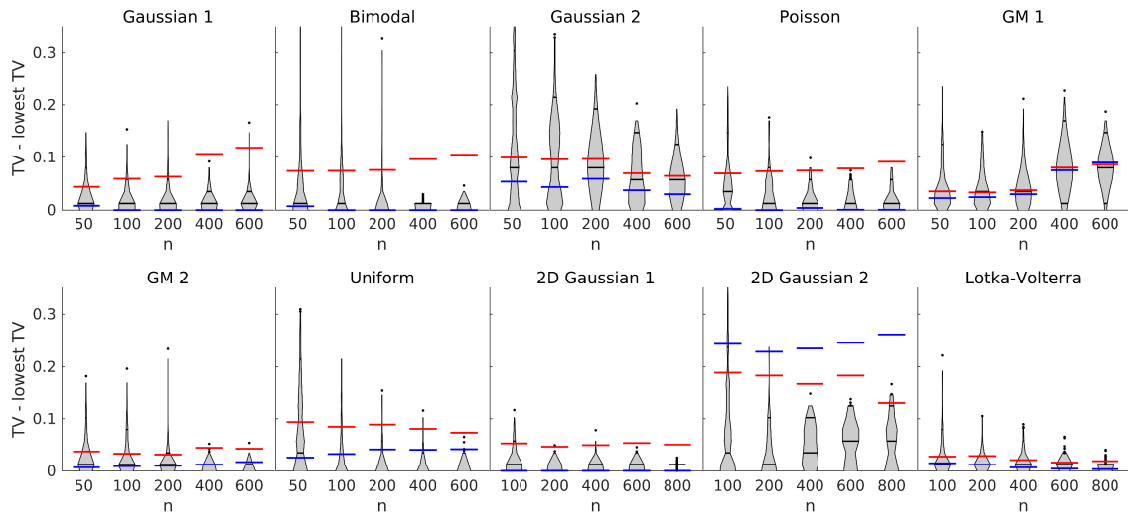


Figure 8: Results of the GP model selection using the mlpd utility. See the caption of the Figure 6 in the main text for details.

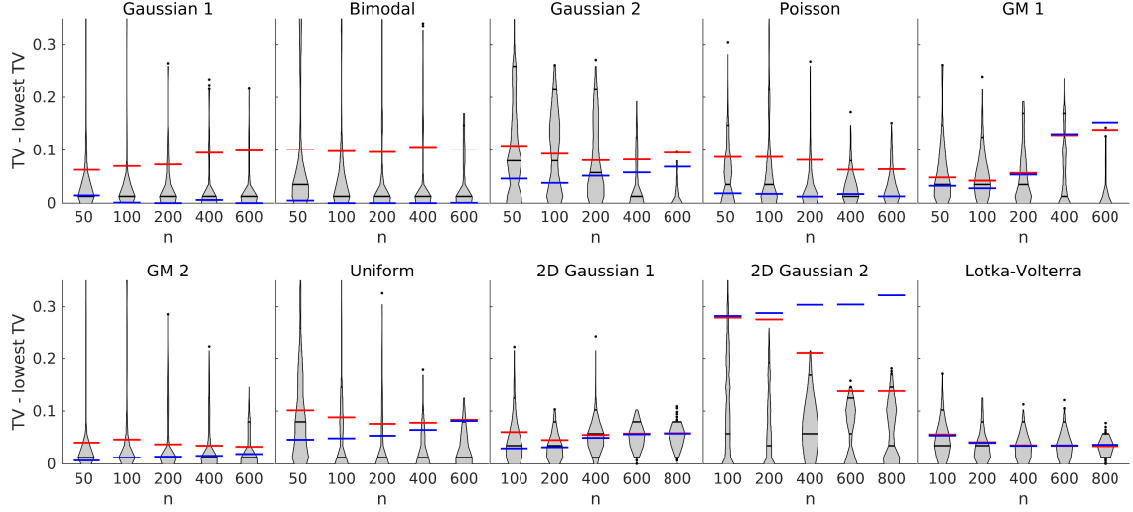


Figure 9: Results of the GP model selection using the classifier utility. The experiments are the same as in Figure 8 expect that comparisons are made to the true posterior.

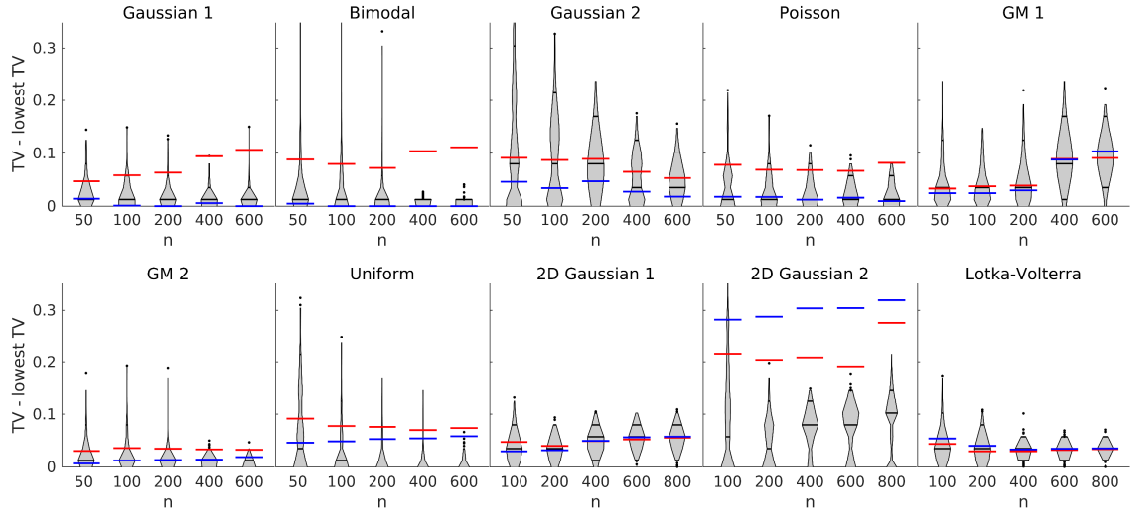


Figure 10: Results of the GP model selection using the mlpd utility. The experiments are the same as in Figure 8 expect that comparisons are made to the true posterior.

A.2 Posterior estimates compared to the true posterior

Tables 4 and 5 contain the results of the experiments in the main text, but the baseline is the true posterior instead of the ABC posterior. Using the true posterior as a baseline does not change the main conclusions although posterior estimates are worse, as expected, because the nonzero threshold also affects the estimates. However, the additional error caused by the threshold is mostly minor in 1d test problems but in 2d test problems (especially with Lotka-Volterra model in which the discrepancy was a sum of squared errors of individual data points), the posterior estimates are clearly worse.

	n=50			n=100			n=200			n=400			n=600		
	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt
Gaussian 1:															
GP	0.19	0.09	0.08	0.22	0.09	0.06	0.22	0.09	0.04	0.22	0.15	0.04	0.22	0.16	0.03
GP in.dep.	0.19	0.13	0.09	0.18	0.13	0.06	0.19	0.11	0.05	0.20	0.12	0.04	0.23	0.12	0.04
classifier GP	0.41	0.41	0.41	0.34	0.34	0.34	0.19	0.19	0.19	0.11	0.11	0.11	0.09	0.09	0.09
rej. ABC	0.32	0.32	0.32	0.27	0.27	0.27	0.19	0.19	0.19	0.12	0.12	0.12	0.12	0.12	0.12
Bimodal:															
GP	0.22	0.48	0.16	0.21	0.26	0.12	0.22	0.17	0.10	0.22	0.15	0.07	0.22	0.16	0.07
GP in.dep.	0.19	0.60	0.18	0.17	0.40	0.14	0.21	0.27	0.11	0.21	0.24	0.08	0.21	0.22	0.07
classifier GP	0.41	0.41	0.41	0.35	0.35	0.35	0.25	0.25	0.25	0.18	0.18	0.18	0.14	0.14	0.14
rej. ABC	0.46	0.46	0.46	0.40	0.40	0.40	0.26	0.26	0.26	0.21	0.21	0.21	0.17	0.17	0.17
Gaussian 2:															
GP	0.31	0.37	0.26	0.31	0.24	0.25	0.32	0.21	0.24	0.32	0.22	0.23	0.31	0.23	0.22
GP in.dep.	0.48	0.36	0.29	0.45	0.34	0.28	0.43	0.30	0.27	0.41	0.26	0.25	0.40	0.24	0.23
classifier GP	0.35	0.35	0.35	0.34	0.34	0.34	0.25	0.25	0.25	0.18	0.18	0.18	0.15	0.15	0.15
rej. ABC	0.28	0.28	0.28	0.29	0.29	0.29	0.21	0.21	0.21	0.17	0.17	0.17	0.15	0.15	0.15
GM 1:															
GP	0.35	0.32	0.33	0.33	0.30	0.32	0.32	0.30	0.32	0.32	0.23	0.31	0.31	0.20	0.31
GP in.dep.	0.44	0.36	0.35	0.42	0.38	0.33	0.41	0.35	0.32	0.40	0.25	0.31	0.39	0.23	0.30
classifier GP	0.36	0.36	0.36	0.37	0.37	0.37	0.29	0.29	0.29	0.19	0.19	0.19	0.15	0.15	0.15
rej. ABC	0.37	0.37	0.37	0.34	0.34	0.34	0.25	0.25	0.25	0.20	0.20	0.20	0.16	0.16	0.16
GM 2:															
GP	0.21	0.16	0.15	0.20	0.13	0.14	0.21	0.13	0.13	0.22	0.12	0.13	0.22	0.12	0.13
GP in.dep.	0.29	0.20	0.15	0.26	0.18	0.14	0.23	0.17	0.13	0.24	0.17	0.12	0.26	0.16	0.12
classifier GP	0.38	0.38	0.38	0.33	0.33	0.33	0.21	0.21	0.21	0.16	0.16	0.16	0.15	0.15	0.15
rej. ABC	0.34	0.34	0.34	0.30	0.30	0.30	0.22	0.22	0.22	0.18	0.18	0.18	0.16	0.16	0.16
Uniform															
GP	0.28	0.19	0.17	0.29	0.20	0.16	0.29	0.19	0.17	0.29	0.20	0.17	0.29	0.20	0.17
GP in.dep.	0.27	0.20	0.17	0.24	0.20	0.13	0.21	0.20	0.12	0.19	0.20	0.12	0.17	0.20	0.11
classifier GP	0.44	0.44	0.44	0.35	0.35	0.35	0.23	0.23	0.23	0.14	0.14	0.14	0.10	0.10	0.10
rej. ABC	0.35	0.35	0.35	0.33	0.33	0.33	0.25	0.25	0.25	0.21	0.21	0.21	0.19	0.19	0.19
Poisson:															
GP	0.21	0.13	0.09	0.20	0.09	0.08	0.20	0.08	0.07	0.22	0.09	0.07	0.22	0.10	0.07
GP in.dep.	0.20	0.22	0.12	0.22	0.18	0.09	0.25	0.16	0.07	0.24	0.14	0.07	0.25	0.14	0.08
classifier GP	0.34	0.34	0.34	0.29	0.29	0.29	0.15	0.15	0.15	0.09	0.09	0.09	0.09	0.09	0.09
rej. ABC	0.27	0.27	0.27	0.24	0.24	0.24	0.18	0.18	0.18	0.12	0.12	0.12	0.11	0.11	0.11

Table 4: Results for the 1D toy examples. The numeric values describe the TV distance between the estimated and the corresponding true posterior density.

	n=100			n=200			n=400			n=600			n=800		
	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt
2D Gaussian 1:															
GP	0.35	0.19	0.19	0.34	0.15	0.18	0.33	0.11	0.16	0.33	0.10	0.16	0.33	0.10	0.15
GP in.dep.	0.27	0.20	0.17	0.26	0.17	0.16	0.26	0.15	0.15	0.28	0.14	0.14	0.28	0.14	0.14
classifier GP	0.69	0.69	0.69	0.70	0.70	0.70	0.29	0.29	0.29	0.21	0.21	0.21	0.19	0.19	0.19
rej. ABC	0.43	0.43	0.43	0.35	0.35	0.35	0.30	0.30	0.30	0.27	0.27	0.27	0.25	0.25	0.25
2D Gaussian 2:															
GP	0.62	0.44	0.56	0.62	0.39	0.54	0.61	0.35	0.52	0.61	0.34	0.52	0.61	0.33	0.52
GP in.dep.	0.72	0.30	0.34	0.72	0.25	0.33	0.56	0.22	0.32	0.56	0.21	0.32	0.56	0.20	0.31
classifier GP	0.70	0.70	0.70	0.71	0.71	0.71	0.44	0.44	0.44	0.28	0.28	0.28	0.24	0.24	0.24
rej. ABC	0.45	0.45	0.45	0.40	0.40	0.40	0.34	0.34	0.34	0.32	0.32	0.32	0.31	0.31	0.31
Lotka-Volterra:															
GP	0.58	0.52	0.55	0.57	0.49	0.52	0.56	0.48	0.51	0.56	0.47	0.51	0.56	0.47	0.50
GP in.dep.	0.65	0.50	0.53	0.56	0.48	0.51	0.55	0.47	0.50	0.55	0.47	0.50	0.55	0.47	0.50
classifier GP	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.79	0.54	0.54	0.54	0.51	0.51	0.51
rej. ABC	0.65	0.65	0.65	0.59	0.59	0.59	0.55	0.55	0.55	0.56	0.56	0.56	0.54	0.54	0.54

Table 5: Results for the 2D toy examples. The numeric values describe the TV distance between the estimated and the corresponding true posterior density.

A.3 Selection of the threshold

The 0.05th quantile of the simulated discrepancies was chosen for the experiments shown in the main text and for the additional experiments in Sections A.1 and A.2. However, we also repeated the experiments with the 0.01th quantile and the corresponding results are shown in Tables 6 and 7. The ABC posterior is used as the baseline here. In general, the results are similar to those with the 0.05th quantile. However, as expected, the classification GP and ABC rejection sampler methods perform much worse due to the smaller number of discrepancy realisations below the threshold. The computations were also repeated with the 0.1th quantile and the results were generally similar to those using the 0.05th quantile, except that the error was larger when compared to the true posterior. Consequently, these results are not reported here.

	n=50			n=100			n=200			n=400			n=600		
	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt
Gaussian 1:															
GP	0.18	0.20	0.08	0.21	0.23	0.06	0.21	0.24	0.05	0.21	0.32	0.04	0.21	0.32	0.03
GP in.dep.	0.20	0.28	0.10	0.18	0.31	0.07	0.19	0.29	0.06	0.20	0.30	0.05	0.22	0.30	0.05
classifier GP	0.51	0.51	0.51	0.56	0.56	0.56	0.56	0.56	0.56	0.52	0.52	0.52	0.39	0.39	0.39
rej. ABC	0.60	0.60	0.60	0.60	0.60	0.60	0.37	0.37	0.37	0.26	0.26	0.26	0.20	0.20	0.20
Bimodal:															
GP	0.21	0.53	0.16	0.21	0.45	0.13	0.21	0.31	0.11	0.21	0.31	0.09	0.21	0.33	0.08
GP in.dep.	0.18	0.68	0.20	0.17	0.60	0.15	0.21	0.49	0.12	0.21	0.46	0.10	0.21	0.43	0.09
classifier GP	0.44	0.44	0.44	0.44	0.44	0.44	0.46	0.46	0.46	0.43	0.43	0.43	0.38	0.38	0.38
rej. ABC	0.47	0.47	0.47	0.47	0.47	0.47	0.48	0.48	0.48	0.44	0.44	0.44	0.32	0.32	0.32
Gaussian 2:															
GP	0.32	0.46	0.27	0.32	0.31	0.25	0.32	0.28	0.25	0.32	0.30	0.23	0.32	0.31	0.23
GP in.dep.	0.48	0.48	0.35	0.45	0.50	0.35	0.44	0.45	0.32	0.42	0.43	0.30	0.41	0.41	0.30
classifier GP	0.45	0.45	0.45	0.43	0.43	0.43	0.47	0.47	0.47	0.47	0.47	0.47	0.44	0.44	0.44
rej. ABC	0.48	0.48	0.48	0.48	0.48	0.48	0.36	0.36	0.36	0.29	0.29	0.29	0.27	0.27	0.27
GM 1:															
GP	0.34	0.32	0.32	0.32	0.31	0.31	0.31	0.32	0.31	0.31	0.30	0.30	0.30	0.29	0.29
GP in.dep.	0.43	0.48	0.34	0.41	0.54	0.32	0.40	0.49	0.31	0.39	0.43	0.29	0.38	0.41	0.29
classifier GP	0.42	0.42	0.42	0.43	0.43	0.43	0.43	0.43	0.43	0.42	0.42	0.42	0.38	0.38	0.38
rej. ABC	0.43	0.43	0.43	0.43	0.43	0.43	0.39	0.39	0.39	0.34	0.34	0.34	0.29	0.29	0.29
GM 2:															
GP	0.19	0.18	0.13	0.19	0.22	0.12	0.19	0.22	0.12	0.21	0.22	0.12	0.21	0.22	0.12
GP in.dep.	0.28	0.31	0.14	0.25	0.34	0.13	0.22	0.30	0.12	0.23	0.28	0.11	0.25	0.27	0.10
classifier GP	0.43	0.43	0.43	0.46	0.46	0.46	0.47	0.47	0.47	0.47	0.47	0.47	0.33	0.33	0.33
rej. ABC	0.49	0.49	0.49	0.49	0.49	0.49	0.36	0.36	0.36	0.28	0.28	0.28	0.25	0.25	0.25
Uniform															
GP	0.27	0.35	0.16	0.28	0.38	0.16	0.28	0.37	0.17	0.28	0.38	0.17	0.28	0.38	0.17
GP in.dep.	0.27	0.31	0.17	0.23	0.33	0.13	0.21	0.35	0.13	0.19	0.39	0.12	0.18	0.38	0.12
classifier GP	0.45	0.45	0.45	0.49	0.49	0.49	0.51	0.51	0.51	0.51	0.51	0.51	0.48	0.48	0.48
rej. ABC	0.51	0.51	0.51	0.51	0.51	0.51	0.42	0.42	0.42	0.33	0.33	0.33	0.30	0.30	0.30
Poisson:															
GP	0.20	0.21	0.10	0.19	0.25	0.08	0.20	0.25	0.07	0.21	0.29	0.06	0.21	0.31	0.07
GP in.dep.	0.20	0.39	0.14	0.21	0.40	0.12	0.26	0.38	0.09	0.25	0.37	0.10	0.26	0.38	0.10
classifier GP	0.45	0.45	0.45	0.50	0.50	0.50	0.38	0.38	0.38	0.28	0.28	0.28	0.18	0.18	0.18
rej. ABC	0.57	0.57	0.57	0.43	0.43	0.43	0.26	0.26	0.26	0.19	0.19	0.19	0.16	0.16	0.16

Table 6: Results for the 1D toy examples when the threshold is 0.01th quantile. The numeric values describe the TV distance between the estimated and the corresponding ABC posterior density.

	n=100			n=200			n=400			n=600			n=800		
	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt	se	log	sqrt
2D Gaussian 1:															
GP	0.29	0.29	0.12	0.28	0.28	0.09	0.27	0.28	0.07	0.27	0.28	0.07	0.27	0.29	0.06
GP in.dep.	0.24	0.41	0.16	0.24	0.33	0.11	0.23	0.28	0.09	0.25	0.27	0.08	0.25	0.27	0.07
classifier GP	0.65	0.65	0.65	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.67
rej. ABC	0.68	0.68	0.68	0.68	0.68	0.68	0.37	0.37	0.37	0.34	0.34	0.34	0.31	0.31	0.31
2D Gaussian 2:															
GP	0.59	0.38	0.52	0.59	0.28	0.49	0.58	0.25	0.47	0.57	0.25	0.46	0.57	0.25	0.46
GP in.dep.	0.69	0.50	0.41	0.67	0.39	0.37	0.59	0.35	0.36	0.59	0.31	0.37	0.59	0.29	0.35
classifier GP	0.68	0.68	0.68	0.68	0.68	0.68	0.69	0.69	0.69	0.69	0.69	0.69	0.69	0.68	0.68
rej. ABC	0.70	0.70	0.70	0.70	0.70	0.70	0.41	0.41	0.41	0.36	0.36	0.36	0.32	0.32	0.32
Lotka-Volterra:															
GP	0.28	0.24	0.25	0.24	0.20	0.20	0.21	0.18	0.18	0.20	0.17	0.16	0.19	0.16	0.14
GP in.dep.	0.53	0.31	0.30	0.29	0.27	0.24	0.25	0.23	0.20	0.24	0.22	0.18	0.22	0.20	0.16
classifier GP	0.59	0.59	0.59	0.60	0.60	0.60	0.59	0.59	0.59	0.59	0.59	0.59	0.59	0.59	0.59
rej. ABC	0.60	0.60	0.60	0.60	0.60	0.60	0.35	0.35	0.35	0.31	0.31	0.31	0.29	0.29	0.29

Table 7: Results for the 2D toy examples when the threshold is the 0.01th quantile. The numeric values describe the TV distance between the estimated and the corresponding ABC posterior density.

B Bivariate posterior marginals for the bacterial genomic model

The bivariate marginal posteriors for the bacterial genomic model described in Section 3.3 of the main text are shown in Figure 11.

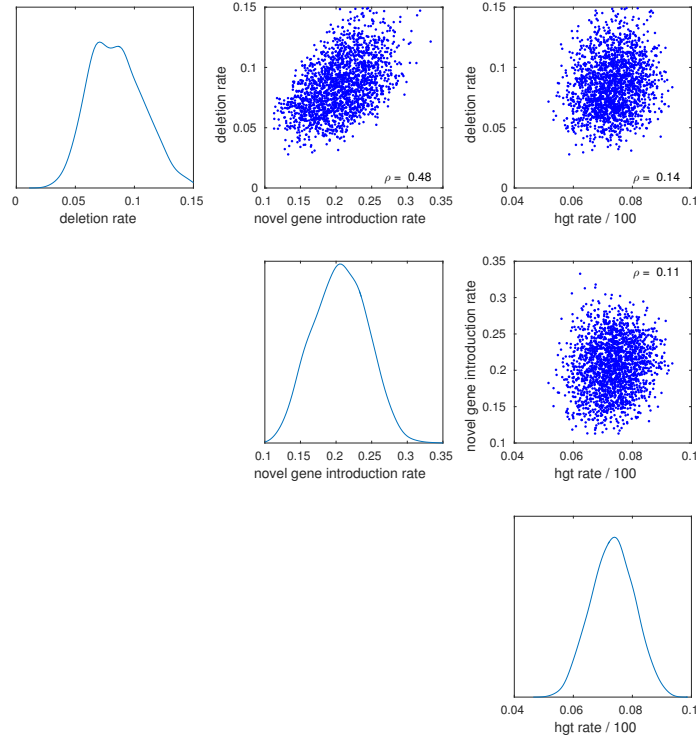


Figure 11: Estimated posterior distribution for the bacterial genomic model. The value of ρ is the (Pearson) correlation coefficient between the corresponding parameters of the model.