Statistical Inference of Intractable Generative Models via Classification

Michael U. Gutmann, Ritabrata Dutta, Samuel Kaski, and Jukka Corander

Abstract—Increasingly complex generative models are being used across the disciplines as they allow for realistic characterization of data, but a common difficulty with them is the prohibitively large computational cost to perform likelihood-based statistical inference. We consider here a likelihood-free framework where inference is done by identifying parameter values which generate simulated data adequately resembling the observed data. A major difficulty is how to measure the discrepancy between the simulated and observed data. Transforming the original problem into a problem of classifying the data into simulated versus observed, we find that classification accuracy can be used to assess the discrepancy. The complete arsenal of classification methods becomes thereby available for inference of intractable generative models.

Index Terms—intractable likelihood, latent variables, simulator-based models, approximate Bayesian computation

1 Introduction

Uncertainty and randomness are an integral part of our life – sometimes to our dismay sometimes to our delight. In science and engineering, probabilistic modeling and statistics provide a principled framework for dealing with uncertainty and for making inferences from data. The likelihood function plays a central role by quantifying to which extent some values of the model parameters are consistent with the observed data. For complex models, however, evaluating the likelihood function can be computationally very costly, which often prevents its use in practice. This paper is about statistical inference of generative models whose likelihood-function cannot be computed in a reasonable time.[1]

A generative model is here defined rather generally as a parametrized probabilistic mechanism which specifies how the data are generated. It is usually implemented as a computer program which takes a state of the random number generator and some values of the model parameters $\boldsymbol{\theta}$ as input and which returns simulated data $\mathbf{Y}_{\boldsymbol{\theta}}$ as output. The mapping from parameter $\boldsymbol{\theta}$ to simulated data $\mathbf{Y}_{\boldsymbol{\theta}}$ is stochastic and running the computer program for different states of the random number generator corresponds to sampling from the model. Generative models are also known as simulator- or simulation-based models [1], implicit models [2], and are closely related to probabilistic programs [3]. The scope of their applicability is extremely wide ranging from genetics and ecology [4] to economics [5], astrophysics [6] and computer vision [7].

- *MUG is with the Dept of Mathematics and Statistics, University of Helsinki, the Dept of Information and Computer Science, Aalto University, and the Helsinki Institute for Information Technology HIIT.*
- *RD is with the Dept of Information and Computer Science, Aalto University, and the Helsinki Institute for Information Technology HIIT.*
- *SK is with the Dept of Information and Computer Science, Aalto University, and the Helsinki Institute for Information Technology HIIT.*
- *JK is with the Dept of Mathematics and Statistics, University of Helsinki, and the Helsinki Institute for Information Technology HIIT.*

1. Preliminary results were presented in the form of a poster, titled "Classifier ABC", at the MCMSki IV meeting in Chamonix, France, in January 2014.

A disadvantage of complex generative models is the difficulty of performing inference with them: evaluating the likelihood function involves computing the probability of the observed data $\mathbf{X}$ as function of the model parameters $\boldsymbol{\theta}$, which cannot be done analytically or computationally within practical time limits. As generative models are widely applicable, solutions have emerged in multiples fields to perform "likelihood-free" inference, that is, inference which does not make use of the likelihood function. Approximate Bayesian computation (ABC), for instance, stems from research in genetics [8], [9], [10], [11], while the method of simulated moments [12], [13] and indirect inference [5], [14] come from econometrics. The latter methods are traditionally used in a classical inference framework while ABC has its roots in Bayesian inference, but the boundaries have started to blur [15]. Despite their differences, the methods all share the basic idea to perform inference about $\boldsymbol{\theta}$ by identifying values which generate simulated data $\mathbf{Y}_{\boldsymbol{\theta}}$ that resemble the observed data $\mathbf{X}$.

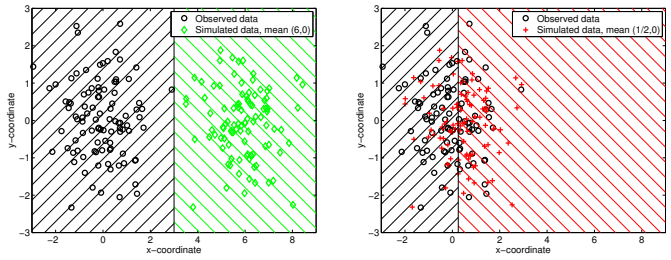
The discrepancy between the simulated and observed data is typically measured by reducing each data set to a vector of summary statistics and measuring the distance between them. Both the distance function used and the summary statistics are critical for the success of the inference procedure. Traditionally, researchers choose the two quantities subjectively, relying on expert knowledge about the observed data. The goal of the paper is to show that the complete arsenal of classification methods is at our disposal to measure the discrepancy, and thus to perform inference for intractable generative models.

2 Motivation for Using Classification

The paper is motivated by the observation that distinguishing two data sets which were generated with very different values of $\boldsymbol{\theta}$ is usually easier than discriminating data which were generated with similar values. We thus propose to use the discriminability (classifiability) of the observed and simulated data as discrepancy measure in likelihood-free inference.

We visualize the motivating idea in Figure 1 for the inference of the mean $\boldsymbol{\theta}$ of a bivariate Gaussian with identity covariance matrix. The observed data $\mathbf{X}$, shown with black circles, were generated with mean $\boldsymbol{\theta}^o$ equal to zero. Figure 1a shows that data $\mathbf{Y}_{\boldsymbol{\theta}}$ simulated with mean $\boldsymbol{\theta} = (6, 0)$ can be easily distinguished from $\mathbf{X}$. The indicated classification rule yields an accuracy of 100%. In Figure 1b, on the other hand, the data were simulated with $\boldsymbol{\theta} = (1/2, 0)$ and distinguishing such data from $\mathbf{X}$ is much more difficult; the best classification rule only yields 58% correct assignments. Moreover, if the data were simulated with $\boldsymbol{\theta} = \boldsymbol{\theta}^o$, the classification task could not be solved significantly above chance-level. This suggests that we can perform likelihood-free inference by identifying parameters which yield chance-level discriminability only.

We will here not be advocating the use of a particular classifier nor develop a highly specific algorithm. We will demonstrate the proposed approach with a range of different classifiers, but the classifiers used will not be very sophisticated either. Our focus will be on communicating the new inference principle and the tight connection between likelihood-free inference and classification – two fields of research which were thought to be rather distant to date.



(a) High discriminability (b) Low discriminability

Fig. 1: Discriminability as discrepancy measure. The observed data $\mathbf{X}$ are shown as black circles and were generated with mean $\boldsymbol{\theta}^o = (0, 0)$. The hatched areas indicate the Bayes classification rules. (a) Simulated data $\mathbf{Y}_{\boldsymbol{\theta}}$ (green diamonds) were generated with $\boldsymbol{\theta} = (6, 0)$. (b) $\mathbf{Y}_{\boldsymbol{\theta}}$ (red crosses) were generated with $\boldsymbol{\theta} = (1/2, 0)$. As $\boldsymbol{\theta}$ approaches $\boldsymbol{\theta}^o$, the discriminability (best classification accuracy) of $\mathbf{X}$ and $\mathbf{Y}_{\boldsymbol{\theta}}$ drops. We propose to use the discriminability as discrepancy measure for likelihood-free inference.

The remaining parts of the paper are structured as follows: In Section 3, we flesh out the motivating idea. We then show in Sections 4 and 5 how classification allows us to perform statistical inference of generative models in both a classical and Bayesian framework. The approach will be validated on continuous, binary, discrete, and time series data where ground truth is known. In Section 6, we apply the methodology to real data. Section 7 discusses the proposed approach and related work. We there also discuss how it relates to perception in artificial and biological systems. Section 8 concludes the paper.

3 Measuring Discrepancy via Classification

We formulate the problem of measuring the discrepancy between the observed data $\mathbf{X}$ and the simulated data $\mathbf{Y}_{\boldsymbol{\theta}}$ as a classification problem. The classification problem is set up by extracting n features (covariates) $\mathbf{x}_i$ and $\mathbf{y}_i$ from $\mathbf{X}$ and $\mathbf{Y}_{\boldsymbol{\theta}}$ and associating them with the class labels 0 and 1, respectively. Unless otherwise stated, the features will equal the elements of $\mathbf{X}$ and $\mathbf{Y}_{\boldsymbol{\theta}}$ but they can also be some transformations of them as seen later in the paper. A data set $\mathcal{D}_{\boldsymbol{\theta}}$ is thereby obtained which consists of $2n$ features together with their class labels,

$$\mathcal{D}_{\boldsymbol{\theta}} = \{(\mathbf{x}_1, 0), \ldots, (\mathbf{x}_n, 0), (\mathbf{y}_1, 1), \ldots, (\mathbf{y}_n, 1)\}. \tag{1}$$

A classification rule h allows us to predict for a feature $\mathbf{u}$, extracted randomly from either $\mathbf{X}$ or $\mathbf{Y}_{\boldsymbol{\theta}}$, its class label $h(\mathbf{u}) \in \{0, 1\}$. The performance of h on $\mathcal{D}_{\boldsymbol{\theta}}$ can be assessed by the classification accuracy CA,

$$\mathrm{CA}(h, \mathcal{D}_{\boldsymbol{\theta}}) = \frac{1}{2n}\left(\sum_{i=1}^{n}[1 - h(\mathbf{x}_i)] + h(\mathbf{y}_i)\right), \tag{2}$$

which is the proportion of correct assignments. The largest classification accuracy on average is achieved by the Bayes classification rule $h^*_{\boldsymbol{\theta}}$ which consists in assigning a feature to $\mathbf{X}$, say, if it is more probable that the feature belongs to $\mathbf{X}$ than to $\mathbf{Y}_{\boldsymbol{\theta}}$ [16], [17]. We denote this largest classification accuracy by $J^*_n(\boldsymbol{\theta})$,

$$J^*_n(\boldsymbol{\theta}) = \mathrm{CA}(h^*_{\boldsymbol{\theta}}, \mathcal{D}_{\boldsymbol{\theta}}). \tag{3}$$

It is an indicator of the discriminability (classifiability) of $\mathbf{X}$ and $\mathbf{Y}_{\boldsymbol{\theta}}$.

In the motivating example in Figure 1, the Bayes classification rule is indicated by the hatched areas. It is seen that its classification accuracy $J^*_n(\boldsymbol{\theta})$ decreases from 100% (perfect classification performance) towards 50% (chance-level performance) as $\boldsymbol{\theta}$ approaches $\boldsymbol{\theta}^o$, the parameter value which was used to generate the observed data $\mathbf{X}$. While this provides an intuitive justification for using $J^*_n(\boldsymbol{\theta})$ as discrepancy measure, an analytical justification will be given in the next section where we show that $J^*_n(\boldsymbol{\theta})$ is related to the total variation distance under mild conditions.

In practice, $J^*_n(\boldsymbol{\theta})$ is not computable because the Bayes classification rule $h^*_{\boldsymbol{\theta}}$ involves the probability distribution of the data which is unknown in the first place. But the classification literature provides a wealth of methods to learn an approximation $\hat{h}_{\boldsymbol{\theta}}$ of the Bayes classification rule, and $J^*_n(\boldsymbol{\theta})$ can be estimated via cross-validation [16], [17].

We will use of several straightforward methods to obtain $\hat{h}_{\boldsymbol{\theta}}$: linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), L_1-regularized polynomial logistic regression, L_1-regularized polynomial support vector machine (SVM) classification, and a max-aggregation of the above and other methods (max-rule, see Supplementary Material 2.1 for details). These are by no means the only applicable methods. In fact, any method yielding a good approximation of $h^*_{\boldsymbol{\theta}}$ may be chosen; our approach makes the complete arsenal of classification methods available to inference of generative models.

For the approximation of $J^*_n(\boldsymbol{\theta})$, we use K-fold cross-validation where the data $\mathcal{D}_{\boldsymbol{\theta}}$ are divided into K folds of training and validation sets, the different validation sets being disjoint. The training sets are used to learn the classification rules $\hat{h}^k_{\boldsymbol{\theta}}$ by any of the methods above, and the validation sets $\mathcal{D}^k_{\boldsymbol{\theta}}$ are used to measure their performances $\mathrm{CA}(\hat{h}^k_{\boldsymbol{\theta}}, \mathcal{D}^k_{\boldsymbol{\theta}})$. The average classification accuracy on the validation sets, $J_n(\boldsymbol{\theta})$,

$$J_n(\boldsymbol{\theta}) = \frac{1}{K}\sum_{k=1}^{K}\mathrm{CA}(\hat{h}^k_{\boldsymbol{\theta}}, \mathcal{D}^k_{\boldsymbol{\theta}}), \tag{4}$$

approximates $J^*_n(\boldsymbol{\theta})$, and is used as computable measure of discrepancy between $\mathbf{X}$ and $\mathbf{Y}_{\boldsymbol{\theta}}$.

We next illustrate on a range of different kinds of data that most of the different classification methods yield equally good approximations of $J^*_n(\boldsymbol{\theta})$ for large sample sizes. Continuous data (drawn from a univariate Gaussian distribution of variance one), binary data (from a Bernoulli distribution), count data (from a Poisson distribution), and time-series data (from a zero mean moving average model of order one) are considered. For the first three data sets, the unknown parameter is the mean, and for the moving average model, the lag coefficient is the unknown quantity (see Supplementary Material 2.2 for the model specifications). Unlike for the other three data sets, the data points from the moving average model are not statistically independent, as the lag coefficient affects the correlation between two consecutive time points x_t and x_{t+1}. For the classification, we treated each pair (x_t, x_{t+1}) as a feature.

Figure 2 shows that for the Gaussian, Bernoulli, and Poisson data, all considered classification methods perform as well as the Bayes classification rule (BCR), yielding discrepancy

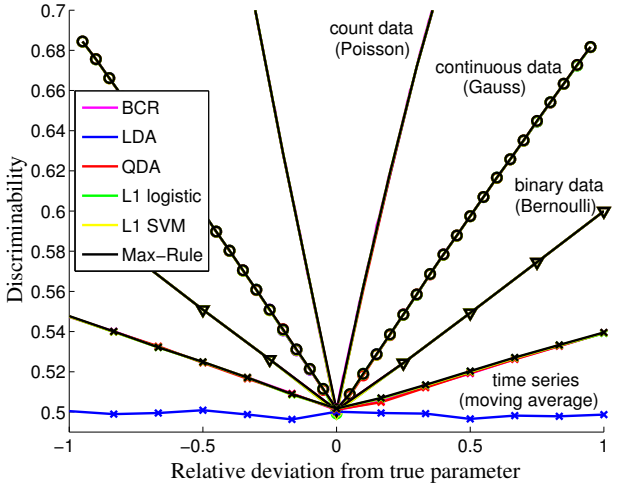


Fig. 2: Comparison of the classification accuracy of the Bayes and the learned classification rules for large sample sizes (n = 100,000). The symmetric curves depict J_n and J_n^* as a function of the relative deviation of the model parameter from the true data generating parameter. As the curves of the different methods are indistinguishable, quadratic discriminant analysis (QDA), L_1-regularized polynomial logistic regression (L1 logistic), L_1-regularized polynomial support vector machine classification (L1 SVM), and a max-combination of these and other methods (Max-Rule) perform as well as the Bayes classification rule, which assumes the true distributions to be known (BCR). For linear discriminant analysis (LDA), this holds with the exception of the moving average model.

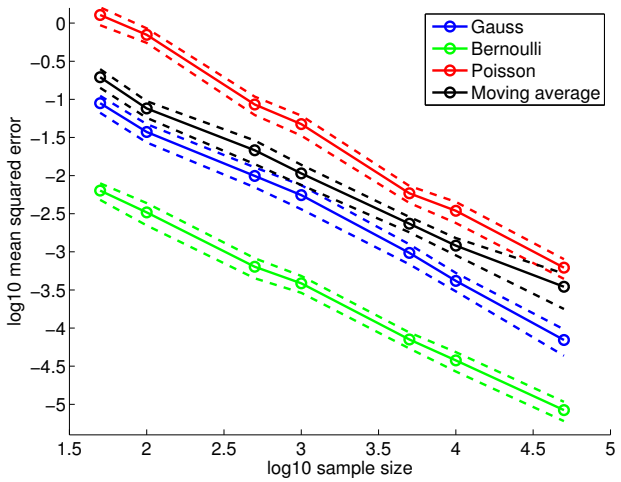


Fig. 3: Empirical evidence for consistency. The figure shows the mean squared estimation error $\mathrm{E}[||\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^o||^2]$ for the examples in Figure 2 as a function of the sample size n (solid lines, circles). The mean was computed as an average over 100 outcomes. The dashed lines depict the mean $\pm$ 2 standard errors. The linear trend on the log-log scale suggests convergence in quadratic mean, and hence consistency of the estimator $\hat{\boldsymbol{\theta}}_n$. The results are for L_1-regularized logistic regression, see Supplementary Material 4 for the other classification methods.

measures $J_n(\boldsymbol{\theta})$ which are practically identical to $J_n^*(\boldsymbol{\theta})$. The same holds for the moving average model, with the exception of LDA. The reason is that LDA is not sensitive to the correlation between x_t and x_{t+1} which would be needed to discover the value of the lag coefficient. In other words, the Bayes classification rule $h^*_{\boldsymbol{\theta}}$ is outside the family of possible classification rules learned by LDA.

The examples show that classification can be used to identify the data generating parameter value $\boldsymbol{\theta}^o$ by minimizing $J_n(\boldsymbol{\theta})$. Further examples are given in Supplementary Material 3. The derivation of conditions which guarantee the identification of $\boldsymbol{\theta}^o$ via classification in general is the topic of the next section.

4 Classical Inference via Classification

In this section, we consider the task of finding the single best parameter value. This can be the primary goal or only the first step before computing the posterior distribution, which will be considered in the following section. In our context, the best parameter value is the value for which the simulated data $\mathbf{Y}_{\boldsymbol{\theta}}$ are the least distinguishable from the observed data $\mathbf{X}$, that is, the parameter $\hat{\boldsymbol{\theta}}_n$ which minimizes J_n,

$$\hat{\boldsymbol{\theta}}_n = \mathrm{argmin}_{\boldsymbol{\theta}} J_n(\boldsymbol{\theta}). \tag{5}$$

We now show that $\hat{\boldsymbol{\theta}}_n$ is a consistent estimator: Assuming that the observed data $\mathbf{X}$ equal some $\mathbf{Y}_{\boldsymbol{\theta}^o}$, generated with unknown parameter $\boldsymbol{\theta}^o$, conditions are given under which $\hat{\boldsymbol{\theta}}_n$ converges to $\boldsymbol{\theta}^o$ in probability as the sample size n increases. Figure 3 provides motivating evidence for consistency of $\hat{\boldsymbol{\theta}}_n$.

The proposition below lists two conditions. The first one is related to convergence of frequencies to expectations (law of large numbers), the second to the ability to learn the Bayes classification rule more accurately as the sample size increases. The proposition is proved in Supplementary Material 1. Some basic assumptions are made: The $\mathbf{x}_i$ are assumed to have the marginal probability measure $\mathrm{P}_{\boldsymbol{\theta}^o}$ and the $\mathbf{y}_i$ the marginal probability measure $\mathrm{P}_{\boldsymbol{\theta}}$ for all i, which amounts to a weak stationarity assumption. The stationarity assumption does not rule out statistical dependencies between the data points; time-series data, for example, are allowed. We also assume that the parametrization of $\mathrm{P}_{\boldsymbol{\theta}}$ is not degenerate, that is, there is a compact set Θ containing $\boldsymbol{\theta}^o$ where $\boldsymbol{\theta} \neq \boldsymbol{\theta}^o$ implies that $\mathrm{P}_{\boldsymbol{\theta}} \neq P_{\boldsymbol{\theta}^o}$.

Proposition 1 *Denote the set of features which the Bayes classification rule $h^*_{\boldsymbol{\theta}}$ classifies as being from the simulated data by $H^*_{\boldsymbol{\theta}}$. The expected discriminability $\mathrm{E}(J_n^*(\boldsymbol{\theta}))$ equals $J(\boldsymbol{\theta})$,*

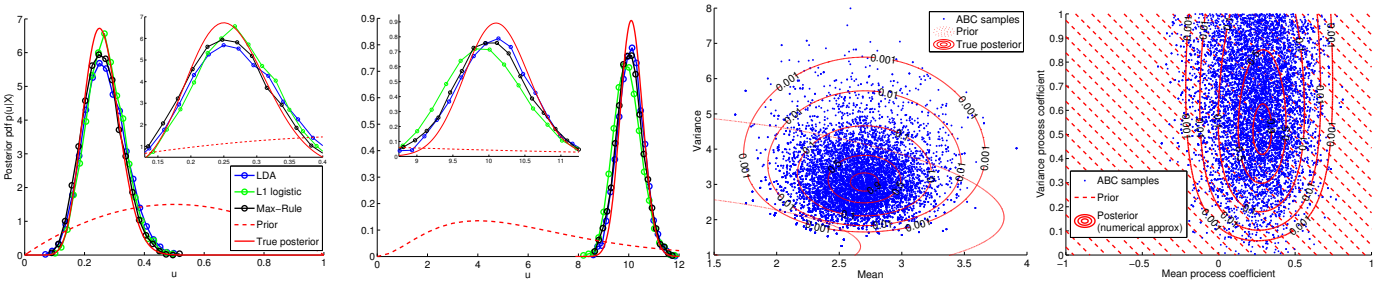
$$J(\boldsymbol{\theta}) = \frac{1}{2} + \frac{1}{2}\left(\mathrm{P}_{\boldsymbol{\theta}}(H^*_{\boldsymbol{\theta}}) - \mathrm{P}_{\boldsymbol{\theta}^o}(H^*_{\boldsymbol{\theta}})\right), \tag{6}$$

and $\hat{\boldsymbol{\theta}}_n$ converges to $\boldsymbol{\theta}^o$ in probability as the sample size n increases, $\hat{\boldsymbol{\theta}}_n \xrightarrow{P} \boldsymbol{\theta}^o$, if

$$\sup_{\boldsymbol{\theta}\in\Theta} |J_n^*(\boldsymbol{\theta}) - J(\boldsymbol{\theta})| \xrightarrow{P} 0 \;\; \textit{and} \;\; \sup_{\boldsymbol{\theta}\in\Theta} |J_n(\boldsymbol{\theta}) - J_n^*(\boldsymbol{\theta})| \xrightarrow{P} 0. \tag{7}$$

The two conditions guarantee that $J_n(\boldsymbol{\theta})$ converges uniformly to $J(\boldsymbol{\theta})$, so that $J(\boldsymbol{\theta})$ is minimized with the minimization of $J_n(\boldsymbol{\theta})$ as n increases. Since $J(\boldsymbol{\theta})$ attains its minimum at $\boldsymbol{\theta}^o$, $\hat{\boldsymbol{\theta}}_n$ converges to $\boldsymbol{\theta}^o$. By definition of $H^*_{\boldsymbol{\theta}}$, $\mathrm{P}_{\boldsymbol{\theta}}(H^*_{\boldsymbol{\theta}}) - \mathrm{P}_{\boldsymbol{\theta}^o}(H^*_{\boldsymbol{\theta}})$ is one half the total variation distance between the two distributions [18, Chapter 3]. The limiting objective $J(\boldsymbol{\theta})$ corresponds thus to a well defined statistical distance between $\mathrm{P}_{\boldsymbol{\theta}}$ and $\mathrm{P}_{\boldsymbol{\theta}^o}$.

The first condition in Equation (7) is about convergence of sample averages to expectations. This is the topic of empirical



(a) binary data (Bernoulli) (b) count data (Poisson) (c) continuous data (Gauss) (d) time series (ARCH)

Fig. 4: Posterior distributions inferred by classifier ABC for binary, count, continuous and time-series data. The results are for 10,000 ABC samples and $n = 50$. For the univariate cases, the samples are summarized as empirical pdfs. For the bivariate cases, scatter plots of the obtained samples are shown (the results are for the max-rule). The numbers on the contours are relative to the maximum of the reference posterior. For the autoregressive conditional heteroskedasticity (ARCH) model, the hatched area indicates the domain of the uniform prior. Supplementary Material 5 contains additional examples and results.

process theory [19] and forms a natural limit of what is studied in this paper. We may only note that by definition of J, convergence will depend on the complexity of the sets $H^*_{\boldsymbol{\theta}}$, $\boldsymbol{\theta} \in \Theta$, and hence the complexity of the Bayes classification rules $h^*_{\boldsymbol{\theta}}$. The condition does not depend on the classification method employed. In other words, the first condition is about the difficulty of the classification problems which need to be solved. The second condition in Equation (7), on the other hand, is about the ability to solve them: The performance of the learned rule needs to approach the performance of the Bayes classification rule as the number of available samples increases. How to best learn such rules and finding conditions which guarantee successful learning is a research area in itself [20].

In Figure 2, LDA did not satisfy the second condition for the moving average data, which can be seen by the chance-level performance for all parameters tested. This failure of LDA suggests a practical means to test whether the second condition holds: We generate data sets with two very different parameters so that it is unlikely that the data sets are similar to each other, and learn to discriminate between them. If the performance is persistently close to chance-level, the Bayes classification rule is likely outside the family of classification rules which the method is able to learn, so that the condition would be violated. Regarding the first condition, the results in Figure 3 suggest that it is satisfied for all four inference problems considered. A practical method to generally verify whether the sample average converges to the expectation seems, however, difficult due to the theoretical nature of the expectation.

5 Bayesian Inference via Classification

We consider next inference of the posterior distribution of $\boldsymbol{\theta}$ in the framework of approximate Bayesian computation (ABC).

ABC comprises several simulation-based methods to obtain samples from the posterior distribution when the likelihood function is not known (for a review, see [21]). ABC algorithms are iterative: The basic steps at each iteration are

1) proposing a parameter $\boldsymbol{\theta}'$,
2) simulating pseudo observed data $\mathbf{Y}_{\boldsymbol{\theta}'}$, and then
3) accepting or rejecting the proposal based on a comparison of $\mathbf{Y}_{\boldsymbol{\theta}'}$ with the real observed data $\mathbf{X}$.

How to actually measure the discrepancy between the observed and the simulated data is a major difficulty in these methods. We here show that J_n can be used as a discrepancy measure in ABC; in the following, we call this approach "classifier ABC." The results were obtained with a sequential Monte Carlo implementation (see Supplementary Material 2.3). The use of J_n in ABC is, however, not restricted to this particular algorithm.

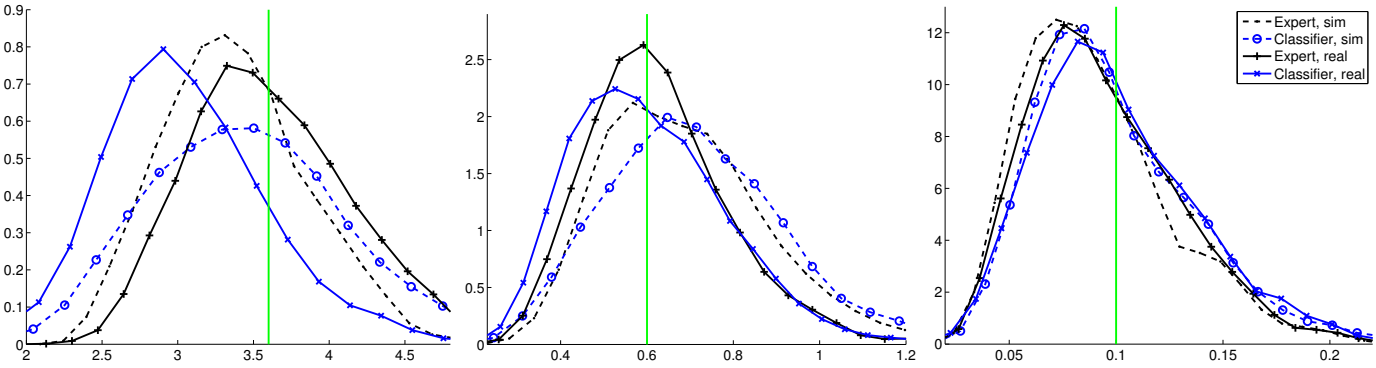
We validated classifier ABC on binary (Bernoulli), count (Poisson), continuous (Gaussian), and time-series (ARCH) data (see Supplementary Material 2.2 for the model details). The true posterior for the autoregressive conditional heteroskedasticity (ARCH) model is not available in closed form. We approximated it using deterministic numerical integration, as detailed in Supplementary Material 2.2.

The inferred empirical posterior probability density functions (pdfs) are shown in Figure 4. There is a good match with the true posterior pdfs or the approximation obtained with deterministic numerical integration. Different classification methods yield different results but the overall performance is rather similar. Regarding computation time, the simpler LDA and QDA tend to be faster than the other classification methods used, with the max-rule being the slowest one. Additional examples as well as links to movies showing the evolution of the posterior samples in the ABC algorithm can be found in Supplementary Material 5.

As a quantitative analysis, we computed the relative error of the posterior means and standard deviations. The results, reported as part of Supplementary Material 5, show that the errors in the posterior mean are within 5% after five iterations of the ABC algorithm for the examples with independent data points. For the time series, where the data points are not independent, a larger error of 15% occurs. The histograms and scatter plots show, however, that the corresponding ABC samples are still very reasonable.

6 Application on Real Data

We next used our discrepancy measure J_n to infer an intractable generative model of bacterial infections in day care centers.



(a) Internal infection parameter β (b) External infection parameter Λ (c) Co-infection parameter θ

Fig. 5: Inferring the individual-based epidemic model with classifier ABC. The results are for real (solid curves) and simulated data (dashed curves). The expert solution with the method by [22] is shown in black (with points and plus markers). The posteriors for classifier ABC are shown in blue (circles and crosses as markers). The green vertical lines indicate $\boldsymbol{\theta}^o$.

6.1 Data and Model

The observed data $\mathbf{X}$ are a random sample of 29 binary matrices of different sizes. Each matrix represents a different day care center at a fixed point of time, and each binary element of a matrix indicates whether a certain attendee is colonized by a particular strain of the bacterium *Streptococcus pneumoniae* or not.

The generative model is individual-based and consists of a continuous-time Markov chain for the transmission dynamics inside a day care center paired with an observation model [22], a more detailed description is provided as Supplementary Material 2.4. The model has three parameters for which uniform priors were assumed: Parameter $\beta \in (0, 11)$ which is related to the probability to be infected by someone inside a day care center, parameter $\Lambda \in (0, 2)$ for the probability of an infection from an outside source, and parameter $\theta \in (0, 1)$ which is related to the probability to be infected with multiple strains. With a slight abuse of notation, we will use $\boldsymbol{\theta} = (\beta, \Lambda, \theta)$ to denote the compound parameter vector.

6.2 Reference Inference Method

Since the likelihood is intractable, the model was inferred with ABC in previous work [22]. The summary statistics were chosen based on epidemiological considerations and the distance function was adapted to the specific problem at hand. A detailed description is given in Supplementary Material 2.4. In brief, the distribution of (a) the strain diversity in the day care centers, (b) the number of different strains circulating, (c) the proportion of individuals which are infected, and (d) the proportion of individuals which are infected with more than one strain were used to compare $\mathbf{X}$ and $\mathbf{Y}_{\boldsymbol{\theta}}$. Inference was performed with a sequential Monte Carlo ABC algorithm with four generations. The corresponding posterior distribution will serve as reference against which we compare the solution by classifier ABC.

6.3 Formulation as Classification Problem

Standard classification methods operate on features in vector form, and hence for likelihood-free inference via classification, the observed matrix-valued data needed to be transformed to feature vectors.

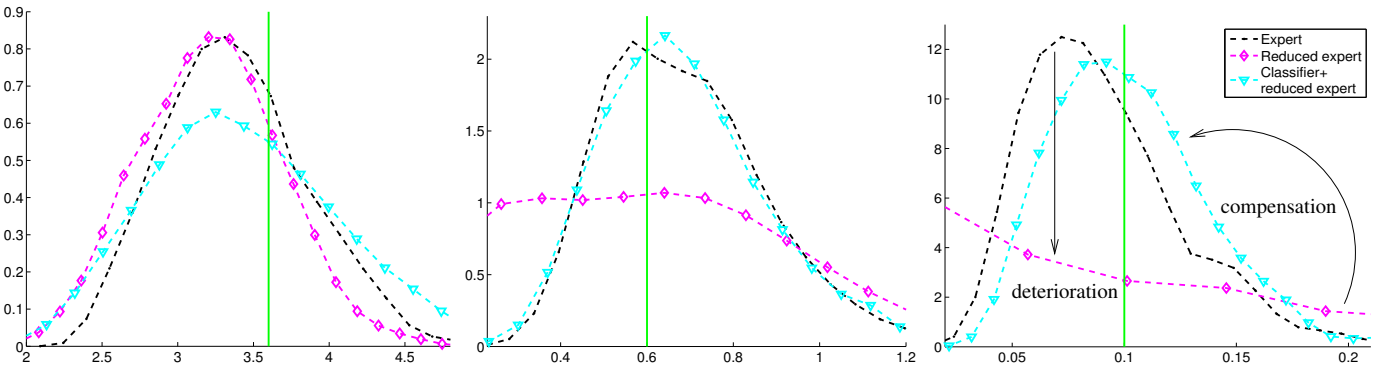
We used simple standard features which reflect the matrix structure and the binary nature of the data: For the matrix-nature of the data, the rank of each matrix and the L_2-norm of the singular values (scaled by the size of the matrix) were used. For the binary nature of the data, we counted the fraction of ones in certain subsets of each matrix and used the average of the counts and their variability as features. The set of rows and the set of columns were used, as well as 100 randomly chosen subsets. Each random subset contained 10% of the elements of a matrix. Since the average of the counts is the same for the row and column subsets (it equals the fraction of all ones in a matrix), only one average was used.

The features $\mathbf{x}_i$ or $\mathbf{y}_i$ in the classification had thus size seven (2 dimensions are for the matrix properties, 3 dimensions for the column and row subsets, and 2 dimensions for the random subsets). Multiple random subsets can be extracted from each matrix. We made use of this to obtain $n = 1{,}000$ features $\mathbf{x}_i$ and $\mathbf{y}_i$. We also ran classifier ABC without random subsets; the classification problems consisted then in discriminating between two data matrices of size 5×29 only. As classification method, we used LDA.

6.4 Inference Results

In ABC, the applicability of a discrepancy measure can be assessed by first performing inference on synthetic data of the same size and structure as the observed data but simulated from the model with known parameter values. The results of such an analysis, including an initial assessment in the framework of classical inference, are reported in Supplementary Material 6. The results show that LDA, the arguably simplest classification method, is suitable to infer the epidemic model.

We then inferred the individual-based epidemic model using classifier ABC. As [22], we used a sequential Monte Carlo ABC algorithm with four generations. The results for classification with random subsets are shown in Figure 5, the results without random subsets are in Supplementary Material 6. The posterior pdfs shown are kernel density estimates (smoothed and scaled histograms) based on 1,000 ABC samples. Classifier ABC yielded similar results as the reference method, on both simulated and real data. For the real data, we note that the posterior mode of β is slightly



(a) Internal infection parameter β (b) External infection parameter Λ (c) Co-infection parameter θ

Fig. 6: Using classifier ABC to compensate for insufficient expert statistics. The results are for data simulated from the epidemic model and are visualized as in Figure 5. The expert solution from that figure is here reproduced for reference. Working with a reduced set of expert statistics affects the posteriors of Λ and θ adversely but classifier ABC is able to compensate (magenta curve with diamonds as markers versus cyan curve with triangles).

smaller for classifier ABC (blue solid versus black solid curve). This shift is not present for classifier ABC without random subsets (see Supplementary Material 6) or for the simulated data. The shift could be due to stochastic variation because we only worked with 1,000 ABC samples. It could, however, also be that the random features picked up some properties of the real data which the other methods are not sensitive to. Inspection of the results in Supplementary Material 6 shows further that omitting the random subsets yields posterior distributions with a larger variance.

The computation time of classifier ABC with LDA was about the same as for the method by [22]: On average, the total time for the data generation and the discrepancy measurement was 28.49 $\pm$ 3.45 seconds for LDA while it was 28.41 $\pm$ 3.45 seconds for the expert method; with 28.4 $\pm$ 3.45 seconds, most of the time was spent on generating data from the epidemic model. Hence, classifier ABC yielded results which are equivalent to the expert solution, from both a statistical and computational point of view.

For the results so far, we did not use expert-knowledge about the problem at hand for classifier ABC. Using discriminability in a classification task as a discrepancy measure is a data-driven approach to assess the similarity between simulated and observed data. But it is not necessarily a black-box approach. Knowledge about the problem at hand can be incorporated when specifying the classification problem. Furthermore, the approach is compatible with summary statistics derived from expert knowledge: Classifier ABC, and more generally the discrepancy measure J_n, is able to incorporate the expert statistics by letting them be features (covariates) in the classification. Combining expert statistics and classifier ABC makes it possible to filter out properties of the model which are either not of interest or known to be wrong. Furthermore, it can be worthwhile if the available expert statistics alone are insufficient to perform ABC, as we illustrate next.

We selected two simple expert statistics of [22], namely the number of different strains circulating and the proportion of infected individuals, and inferred the posteriors with this reduced set of summary statistics, using the method by [22] as before. Figure 6 shows that consequently, the posterior distributions of Λ and θ deteriorated. Combining the insufficient set of summary statistics with classifier ABC, however, led to a recovery of the posteriors. The same happens for classifier ABC without random subsets (Supplementary Material 6).

7 Discussion

Generative models are useful and widely applicable tools for dealing with uncertainty and making inferences from data. The intractability of the likelihood function is, however, often a serious problem in the inference for realistic models. While likelihood-free methods provide a powerful framework for performing inference, a limiting difficulty is the required discrepancy measurement between simulated and observed data. We found that classification can be used to measure the discrepancy. This finding has practical value because it reduces the difficult problem of choosing an appropriate discrepancy measure to a more standard problem where we can leverage a wealth of existing solutions; whenever we can classify, we can do likelihood-free inference. It offers also theoretical value because it reveals that classification can yield consistent likelihood-free inference, and that the two fields of research, which appear very much separated at first glance, are actually tightly connected.

7.1 Summary Statistics versus Features

In the proposed approach, instead of choosing summary statistics and a distance function between them as in the standard approach, we need to choose a classification method and the features. The reader may thus wonder whether we replaced one possibly arbitrary choice with another. The important point is that by choosing a classification method, we only decide about a function space, and not the classification rule itself. The classification rule which is finally used to measure the discrepancy is learned from data and is not specified by the user, which is in stark contrast to the traditional approach based on fixed summary statistics. Moreover, the function space can be chosen using cross-validation, as implemented with our max-rule, which reduces the arbitrariness even more. An example of this can be seen in Figure 2, where the max-rule successfully chose to use other classification methods than LDA for the inference of the moving average model. The

influence of the choice of features is also rather mild, because they only affect the discrepancy measurement via the learned classification rule. This property of the proposed approach allowed us to even use random features in the inference of the epidemic model.

7.2 Related Work

In previous work, regression with the parameters $\boldsymbol{\theta}$ as response variables was used to generate summary statistics from a larger pool of candidates [23], [24], [25]. The common point between this kind of work and our approach is the learning of transformations of the summary statistics and the features, respectively. The criteria which drive the learning are, however, rather different: Since the candidate statistics are a function of the simulated data $\mathbf{Y}_{\boldsymbol{\theta}}$, we may consider the regression to provide an approximate inversion of the data generation process $\boldsymbol{\theta} \mapsto \mathbf{Y}_{\boldsymbol{\theta}}$. In this interpretation, the (Euclidean) distance of the summary statistics is an approximation of the (Euclidean) distance of the parameters. The optimal inversion of the data generating process in a mean squared error sense is the conditional expectation $\mathrm{E}(\boldsymbol{\theta}|\mathbf{Y}_{\boldsymbol{\theta}})$. It is shown in [25] that this conditional expectation is also the optimal summary statistic for $\mathbf{Y}_{\boldsymbol{\theta}}$ if the goal is to infer $\boldsymbol{\theta}^o$ as accurately as possible under a quadratic loss. Transformations based on regression are thus strongly linked to the computation of the distance between the parameters. The reason we learn transformations, on the other hand, is that we would like to approximate $J_n^*(\boldsymbol{\theta})$ well, which is linked to the computation of the total variation distance between the distributions indexed by the parameters.

Classification was recently used in other work on ABC, but in a different manner. Intractable density ratios in Markov chain Monte Carlo algorithms were estimated using tools from classification [26], in particular random forests, and [27] used random forests for model selection by learning to predict the model class from the simulated data instead of computing their posterior probabilities. This is different from using classification to define a discrepancy measure between simulated and observed data, as done here.

A particular classification method, (nonlinear) logistic regression, was used for the estimation of unnormalized models [28], which are models where the probability density (pdf) functions are known up to the normalizing partition function only (see [29] for a review paper and [30], [31] for some generalizations). Likelihood-based inference is intractable for unnormalized models but unlike in the generative models considered here, the shape of the model-pdf is known which can be exploited in the inference.

7.3 Sequential Inference and Relation to Perception

We did not make any specific assumption about the structure of the observed data $\mathbf{X}$ or the generative model. An interesting special case occurs when $\mathbf{X}$ are an element $\mathbf{X}^{(t_0)}$ of a sequence of data sets $\mathbf{X}^{(t)}$ which are observed one after the other, and the generative model is specified accordingly to generate a sequence of simulated data sets.

For inference at t_0, we can distinguish between simulated data which were generated either before or after $\mathbf{X}^{(t_0)}$ are observed: In the former case, the simulated data are predictions about $\mathbf{X}^{(t_0)}$, and after observation of $\mathbf{X}^{(t_0)}$, likelihood-free inference about $\boldsymbol{\theta}$ corresponds to assessing the accuracy of the predictions. That is, the discrepancy measurement converts the predictions of $\mathbf{X}^{(t_0)}$ into inferences of the causes of $\mathbf{X}^{(t_0)}$. In the latter case, each simulated data set can immediately be compared to $\mathbf{X}^{(t_0)}$ which enables efficient iterative identification of parameter values with low discrepancy [32]. That is, the possible causes of $\mathbf{X}^{(t_0)}$ can be explained more accurately with the benefit of hindsight.

Probabilistic modeling and inference play key roles in machine intelligence [33] and robotics [34], and provide a normative theory for a wide range of brain functions [35]. Perception, for instance, has been modeled as (Bayesian) inference based on a "mental" generative model of the world (see [36] for a recent review). In most of the literature, variational approximate inference has been used for intractable generative models, giving rise to the Helmholtz machine [37] and to the free-energy theory of the brain [38]. But other approximate inference methods can be considered as well.

The discussion above highlights similarities between perception and likelihood-free inference or approximate Bayesian computation: It is intuitively sensible that perception would involve prediction of new sensory input given the past, as well as an assessment of the predictions and a refinement of their explanations after arrival of the data. The quality of the inference depends on the quality of the generative model and the quality of the discrepancy assessment. That is, the inference results may only be useful if the generative model of the world is rich enough to produce data resembling the observed data, and if the discrepancy measure can reliably distinguish between the "mentally" generated and the actually observed data.

We proposed to measure the discrepancy via classification, being agnostic about the particular classifier used. It is an open question how to generally best measure the classification accuracy when the data are arriving sequentially. Classifiers are, however, rather naturally part of both biological and artificial perceptual systems. Rapid object recognition, for instance, can be achieved via feedforward multilayer classifiers [39] and there are several techniques to learn representations which facilitate classification [40]. It is thus conceivable that a given classification machinery is used for several purposes, for example to quickly recognize certain objects but also to assess the discrepancy between simulated and observed data.

8 Conclusions

In the paper, we proposed to measure the discrepancy in likelihood-free inference of generative models via classification. We focused on the principle and not on a particular classification method. Some methods may be particularly suited for certain models, where it may also be possible to the measure the discrepancy via the loss function which is used to learn the classification rule instead of the classification accuracy.

Further exploration of the connection between likelihood-free inference and classification is likely to lead to practical improvements in general. Each parameter $\boldsymbol{\theta}$ induces a classification problem. We treated the classification problems separately but they are actually related: First, the observed data $\mathbf{X}$ occur in all the classification problems. Second, the simulated data sets $\mathbf{Y}_{\boldsymbol{\theta}}$ are likely to share some properties

if the parameters are not too different. Taking advantage of the relation between the different classification problems may lead to both computational and statistical gains. In the classification literature, leveraging the solution of one problem to solve another one is generally known as transfer learning [41]. In the same spirit, leveraging transfer learning, or other methods from classification, seems promising to further advance likelihood-free inference.

Acknowledgments

This work was partially supported by ERC grant no. 239784 and the Academy of Finland (Finnish Centre of Excellence in Computational Inference Research COIN, 251170). MUG and RD thank Elina Numminen for providing computer code for the epidemic model.

References

[1] F. Hartig, J. Calabrese, B. Reineking, T. Wiegand, and A. Huth, "Statistical inference for stochastic simulation models – theory and application," *Ecology Letters*, vol. 14, no. 8, pp. 816–827, 2011.

[2] P. Diggle and R. Gratton, "Monte Carlo methods of inference for implicit statistical models," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 46, no. 2, pp. 193–227, 1984.

[3] V. Mansinghka, T. D. Kulkarni, Y. N. Perov, and J. Tenenbaum, "Approximate Bayesian image interpretation using generative probabilistic graphics programs," in *Advances in Neural Information Processing Systems 26 (NIPS)*, 2013.

[4] M. A. Beaumont, "Approximate Bayesian computation in evolution and ecology," *Annual Review of Ecology Evolution and Systematics*, vol. 41, no. 1, pp. 379–406, 2010.

[5] C. Gouriéroux, A. Monfort, and E. Renault, "Indirect inference," *Journal of Applied Econometrics*, vol. 8, no. S1, pp. S85–S118, 1993.

[6] E. Cameron and A. N. Pettitt, "Approximate Bayesian computation for astronomical model analysis: a case study in galaxy demographics and morphological transformation at high redshift," *Monthly Notices of the Royal Astronomical Society*, vol. 425, no. 1, pp. 44–65, 2012.

[7] L. Zhu, Y. Chen, and A. Yuille, "Unsupervised learning of probabilistic grammar-markov models for object categories," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 114–128, 2009.

[8] S. Tavaré, D. Balding, R. Griffiths, and P. Donnelly, "Inferring coalescence times from DNA sequence data," *Genetics*, vol. 145, no. 2, pp. 505–518, 1997.

[9] J. Pritchard, M. Seielstad, A. Perez-Lezaun, and M. Feldman, "Population growth of human Y chromosomes: a study of Y chromosome microsatellites," *Molecular Biology and Evolution*, vol. 16, no. 12, pp. 1791–1798, 1999.

[10] M. Beaumont, W. Zhang, and D. Balding, "Approximate Bayesian computation in population genetics," *Genetics*, vol. 162, no. 4, pp. 2025–2035, 2002.

[11] P. Marjoram, J. Molitor, V. Plagnol, and S. Tavaré, "Markov chain Monte Carlo without likelihoods," *Proceedings of the National Academy of Sciences*, vol. 100, no. 26, pp. 15 324–15 328, 2003.

[12] D. McFadden, "A method of simulated moments for estimation of discrete response models without numerical integration," *Econometrica*, vol. 57, no. 5, pp. 995–1026, 1989.

[13] A. Pakes and D. Pollard, "Simulation and the asymptotics of optimization estimators," *Econometrica*, vol. 57, no. 5, pp. 1027–1057, 1989.

[14] A. Smith, *The New Palgrave Dictionary of Economics*, 2nd ed. Palgrave Macmillan (London), 2008, ch. Indirect Inference.

[15] C. Drovandi, A. Pettitt, and M. Faddy, "Approximate Bayesian computation using indirect inference," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 60, no. 3, pp. 317–337, 2011.

[16] L. Wasserman, *All of statistics*. Springer, 2004.

[17] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2009.

[18] D. Pollard, *A User's Guide to Measure Theoretic Probability*. Cambridge University Press, 2001.

[19] A. van der Vaart and J. Wellner, *Weak Convergence and Empirical Processes*. Springer, 1996.

[20] T. Zhang, "Statistical behavior and consistency of classification methods based on convex risk minimization," *The Annals of Statistics*, vol. 32, no. 1, pp. 56–85, 2004.

[21] J.-M. Marin, P. Pudlo, C. Robert, and R. Ryder, "Approximate Bayesian computational methods," *Statistics and Computing*, vol. 22, no. 6, pp. 1167–1180, 2012.

[22] E. Numminen, L. Cheng, M. Gyllenberg, and J. Corander, "Estimating the transmission dynamics of Streptococcus pneumoniae from strain prevalence data," *Biometrics*, vol. 69, no. 3, pp. 748–757, 2013.

[23] D. Wegmann, C. Leuenberger, and L. Excoffier, "Efficient approximate Bayesian computation coupled with Markov chain Monte Carlo without likelihood," *Genetics*, vol. 182, no. 4, pp. 1207–1218, 2009.

[24] S. Aeschbacher, M. Beaumont, and A. Futschik, "A novel approach for choosing summary statistics in approximate Bayesian computation," *Genetics*, vol. 192, no. 3, pp. 1027–1047, 2012.

[25] 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)*, vol. 74, no. 3, pp. 419–474, 2012.

[26] K. Pham, D. Nott, and S. Chaudhuri, "A note on approximating ABC-MCMC using flexible classifiers," *STAT*, vol. 3, no. 1, pp. 218–227, 2014.

[27] P. Pudlo, J. Marin, A. Estoup, J.-M. Cornuet, M. Gautier, and C. Robert, "ABC model choice via random forests," *ArXiv e-prints*, no. 1406.6288, 2014.

[28] M. Gutmann and A. Hyvärinen, "Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics," *Journal of Machine Learning Research*, vol. 13, pp. 307–361, 2012.

[29] ——, "Estimation of unnormalized statistical models without numerical integration," in *Proceedings of the Sixth Workshop on Information Theoretic Methods in Science and Engineering (WITMSE)*, 2013.

[30] M. Pihlaja, M. Gutmann, and A. Hyvärinen, "A family of computationally efficient and simple estimators for unnormalized statistical models," in *Proceedings of the 26th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2010.

[31] M. Gutmann and J. Hirayama, "Bregman divergence as general framework to estimate unnormalized statistical models," in *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2011.

[32] M. Gutmann and J. Corander, "Bayesian optimization for likelihood-free inference of simulator-based statistical models," *arXiv:1501.03291*, 2015.

[33] Z. Ghahramani, "Probabilistic machine learning and artificial intelligence," *Nature*, vol. 521, no. 7553, pp. 452–459, 2015.

[34] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2006.

[35] A. Pouget, J. M. Beck, W. J. Ma, and P. E. Latham, "Probabilistic brains: knowns and unknowns," *Nature Neuroscience*, vol. 16, no. 9, pp. 1170–1178, 2013.

[36] B. T. Vincent, "A tutorial on Bayesian models of perception," *Journal of Mathematical Psychology*, vol. 66, pp. 103–114, 2015.

[37] P. Dayan, G. Hinton, R. Neal, and R. Zemel, "The Helmholtz machine," *Neural Computation*, vol. 7, no. 5, pp. 889–904, 1995.

[38] K. Friston, "The free-energy principle: a unified brain theory?" *Nature Reviews Neuroscience*, vol. 11, no. 2, pp. 127–138, 2010.

[39] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust object recognition with cortex-like mechanisms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 3, pp. 411–426, 2007.

[40] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[41] S. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

Statistical Inference of Intractable Generative Models via Classification

– Supplementary Materials –

Michael U. Gutmann, Ritabrata Dutta, Samuel Kaski, and Jukka Corander

Contents

List of Figures

List of Tables

1 Proof of Proposition 1

Proposition 1 is proved using an approach based on uniform convergence in probability of J_n to a function J whose minimizer is $\boldsymbol{\theta}^o$ (van der Vaart 1998). The proof has three steps: First, we identify J. Second, we find conditions under which J is minimized by $\boldsymbol{\theta}^o$. Third, we derive conditions which imply that J_n converges to J.

1.1 Definition of J

For validation sets $\mathcal{D}_{\boldsymbol{\theta}}^k$ consisting of $2m$ labeled features $(\mathbf{x}_i^k, 0)$ and $(\mathbf{y}_i^k, 1)$, $i = 1, \ldots, m$, we have by definition of $\mathrm{CA}(h, \mathcal{D}_{\boldsymbol{\theta}})$ in Equation (2) in the main text

$$\mathrm{CA}(\hat{h}_{\boldsymbol{\theta}}^k, \mathcal{D}_{\boldsymbol{\theta}}^k) = \frac{1}{2m}\left(\sum_{i=1}^{m}[1 - \hat{h}_{\boldsymbol{\theta}}^k(\mathbf{x}_i^k)] + \hat{h}_{\boldsymbol{\theta}}^k(\mathbf{y}_i^k)\right) \tag{S1}$$

$$= \frac{1}{2} + \frac{1}{2m}\sum_{i=1}^{m}\hat{h}_{\boldsymbol{\theta}}^k(\mathbf{y}_i^k) - \hat{h}_{\boldsymbol{\theta}}^k(\mathbf{x}_i^k), \tag{S2}$$

so that $J_n(\boldsymbol{\theta})$ in Equation (4) in the main text can be written as

$$J_n(\boldsymbol{\theta}) = \frac{1}{K}\sum_{k=1}^{K}\left(\frac{1}{2} + \frac{1}{2m}\sum_{i=1}^{m}\hat{h}_{\boldsymbol{\theta}}^k(\mathbf{y}_i^k) - \hat{h}_{\boldsymbol{\theta}}^k(\mathbf{x}_i^k)\right) \tag{S3}$$

$$= \frac{1}{2} + \frac{1}{2Km}\sum_{i=1}^{m}\sum_{k=1}^{K}\hat{h}_{\boldsymbol{\theta}}^k(\mathbf{y}_i^k) - \hat{h}_{\boldsymbol{\theta}}^k(\mathbf{x}_i^k). \tag{S4}$$

Each feature is used exactly once for validation since the $\mathcal{D}_{\boldsymbol{\theta}}^k$ are disjoint. We make the simplifying assumption that splitting the original n features into K folds of m features was possible without remainders. We can then order the $\mathbf{y}_i^k$ as

$$\mathbf{y}_1^1, \ldots, \mathbf{y}_m^1, \mathbf{y}_1^2, \ldots, \mathbf{y}_m^2, \mathbf{y}_1^3, \ldots, \mathbf{y}_m^K,$$

and relabel them from 1 to n. Doing the same for the $\mathbf{x}_i^k$, we obtain

$$J_n(\boldsymbol{\theta}) = \frac{1}{2} + \frac{1}{2n}\sum_{i=1}^{n}\hat{h}_{\boldsymbol{\theta}}^{k(i)}(\mathbf{y}_i) - \frac{1}{2n}\sum_{i=1}^{n}\hat{h}_{\boldsymbol{\theta}}^{k(i)}(\mathbf{x}_i). \tag{S5}$$

The function $k(i)$ in the equation indicates to which validation set feature i belonged. If the Bayes classification rule is used instead of the learned $\hat{h}_{\boldsymbol{\theta}}^{k(i)}$, we obtain $J_n^*(\boldsymbol{\theta})$ in Equation (3) in the main text,

$$J_n^*(\boldsymbol{\theta}) = \frac{1}{2} + \frac{1}{2n}\sum_{i=1}^{n}h_{\boldsymbol{\theta}}^*(\mathbf{y}_i) - \frac{1}{2n}\sum_{i=1}^{n}h_{\boldsymbol{\theta}}^*(\mathbf{x}_i). \tag{S6}$$

The function $k(i)$ disappeared because of the weak stationarity assumption in the main text that the marginal distributions of the $\mathbf{x}_i$ and $\mathbf{y}_i$ do not depend on i.

In what follows, it is helpful to introduce the set $H_{\boldsymbol{\theta}}^* = \{\mathbf{u} : h_{\boldsymbol{\theta}}^*(\mathbf{u}) = 1\}$. The normalized sums in (S6) are then the fractions of features which belong to $H_{\boldsymbol{\theta}}^*$. Taking the expectation over $\mathbf{X}$ and $\mathbf{Y}_{\boldsymbol{\theta}}$, using that the expectation over the binary function $h_{\boldsymbol{\theta}}^*$ equals the probability of the set $H_{\boldsymbol{\theta}}^*$,

$$\mathrm{E}(h_{\boldsymbol{\theta}}^*(\mathbf{y}_i)) = \mathrm{P}_{\boldsymbol{\theta}}(H_{\boldsymbol{\theta}}^*), \qquad \mathrm{E}(h_{\boldsymbol{\theta}}^*(\mathbf{x}_i)) = \mathrm{P}_{\boldsymbol{\theta}^o}(H_{\boldsymbol{\theta}}^*), \tag{S7}$$

we obtain the average discriminability $\mathrm{E}(J_n^*(\boldsymbol{\theta})) = J(\boldsymbol{\theta})$,

$$J(\boldsymbol{\theta}) = \frac{1}{2} + \frac{1}{2}\left(\mathrm{P}_{\boldsymbol{\theta}}(H_{\boldsymbol{\theta}}^*) - \mathrm{P}_{\boldsymbol{\theta}^o}(H_{\boldsymbol{\theta}}^*)\right). \tag{S8}$$

The difference between J_n and J is twofold: First, relative frequencies instead of probabilities (expectations) occur. Second, learned classification rules instead of the Bayes classification rule are used. Step 3 of the proof is about conditions which guarantee that J_n converges to J. Prior to that, we first show that J is a meaningful limit, in the sense that its minimization allows to identify $\boldsymbol{\theta}^o$.

Remark. There is an interesting analogy between the objective J_n^* and the log-likelihood: The sum over the $\mathbf{y}_i$ does not depend on the observed data but on $\boldsymbol{\theta}$ and may be considered an analogue to the log-partition function (or an estimate of it). In the same analogy, the sum over the $\mathbf{x}_i$ corresponds to the logarithm of the unnormalized model of the data. The two terms have opposite signs and balance each other as in the methods for unnormalized models reviewed by Gutmann and Hyvärinen (2013).

1.2 Minimization of J

We note that $J(\boldsymbol{\theta}^o) = 1/2$. Since $H^*_{\boldsymbol{\theta}}$ contains only the points which are more probable under $\mathrm{P}_{\boldsymbol{\theta}}$ than under $\mathrm{P}_{\boldsymbol{\theta}^o}$, we have further that $J(\boldsymbol{\theta}) \geq 1/2$. Hence, $\boldsymbol{\theta}^o$ is a minimizer of J. However, $\boldsymbol{\theta}^o$ might not be the only one: Depending on the parametrization, it could be that $\mathrm{P}_{\boldsymbol{\theta}^o} = \mathrm{P}_{\boldsymbol{\theta}}$ for some $\tilde{\boldsymbol{\theta}}$ other than $\boldsymbol{\theta}^o$. We make the identifiability assumption that the $\tilde{\boldsymbol{\theta}}$ are well separated from $\boldsymbol{\theta}^o$ so that there is is a compact subset Θ of the parameter space which contains $\boldsymbol{\theta}^o$ but none of the $\tilde{\boldsymbol{\theta}}$. The above can then be summarized as Proposition 2.

PROPOSITION 2 *$J(\boldsymbol{\theta}^o) = 1/2$ and $J(\boldsymbol{\theta}) > 1/2$ for all other $\boldsymbol{\theta} \in \Theta$.*

Restricting the parameter space to Θ, consistency of $\hat{\boldsymbol{\theta}}_n$ follows from uniform convergence of J_n to J on Θ (van der Vaart 1998, Theorem 5.7).

1.3 Uniform convergence of J_n to J

We show that J_n converges uniformly to J if J^*_n converges to J and if J_n stays close to J^*_n for large n. This splits the convergence problem into two sub-problems with clear meanings which are discussed in the main text.

PROPOSITION 3

$$\textit{If} \quad \sup_{\boldsymbol{\theta}\in\Theta} |J(\boldsymbol{\theta}) - J^*_n(\boldsymbol{\theta})| \xrightarrow{P} 0 \quad \textit{and} \quad \sup_{\boldsymbol{\theta}\in\Theta} |J^*_n(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})| \xrightarrow{P} 0 \quad \textit{then} \quad \sup_{\boldsymbol{\theta}\in\Theta} |J(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})| \xrightarrow{P} 0. \tag{S9}$$

Proof. By the triangle inequality, we have

$$|J(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})| \leq |J(\boldsymbol{\theta}) - J^*_n(\boldsymbol{\theta})| + |J^*_n(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})|, \tag{S10}$$

so that

$$\sup_{\boldsymbol{\theta}\in\Theta} |J(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})| \leq \sup_{\boldsymbol{\theta}\in\Theta} |J(\boldsymbol{\theta}) - J^*_n(\boldsymbol{\theta})| + \sup_{\boldsymbol{\theta}\in\Theta} |J^*_n(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})|, \tag{S11}$$

and hence

$$\mathrm{P}\left(\sup_{\boldsymbol{\theta}\in\Theta} |J(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})| > \epsilon\right) \leq \mathrm{P}\left(\sup_{\boldsymbol{\theta}\in\Theta} |J(\boldsymbol{\theta}) - J^*_n(\boldsymbol{\theta})| + \sup_{\boldsymbol{\theta}\in\Theta} |J^*_n(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})| > \epsilon\right). \tag{S12}$$

It further holds that

$$\mathrm{P}\left(\sup_{\boldsymbol{\theta}\in\Theta} |J(\boldsymbol{\theta}) - J^*_n(\boldsymbol{\theta})| + \sup_{\boldsymbol{\theta}\in\Theta} |J^*_n(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})| > \epsilon\right) \leq \mathrm{P}\left(\sup_{\boldsymbol{\theta}\in\Theta} |J(\boldsymbol{\theta}) - J^*_n(\boldsymbol{\theta})| > \frac{\epsilon}{2}\right) + \mathrm{P}\left(\sup_{\boldsymbol{\theta}\in\Theta} |J^*_n(\boldsymbol{\theta}) - J_n(\boldsymbol{\theta})| > \frac{\epsilon}{2}\right) \tag{S13}$$

which concludes the proof. □

2 MODELS AND ALGORITHMS

This section contains details on the classification methods, the models for continuous, binary, count and time-series data used to test our approach, the ABC algorithm employed, as well as a summary of the epidemic model of Numminen et al. (2013) and their inference method.

2.1 Classification methods

There are many possible classification methods, ranging from traditional logistic regression to more recent deep learning and kernel methods. For an introduction, we refer the reader to the textbooks by Wasserman (2004) and Hastie et al. (2009). We used methods provided by two libraries: For linear and quadratic discriminant analysis (LDA and QDA), matlab's `classify.m` was employed. For L_1 and L_2 regularized polynomial logistic regression and support vector machine (SVM) classification, we used the `liblinear` classification library (Fan et al. 2008), version 1.93, via the matlab interface, with a fixed regularization penalty (we used the default value $C = 1$). The `liblinear` library is for linear classification. Polynomial classification was implemented via polynomial basis expansion (Hastie et al. 2009, Chapter 5). We rescaled the covariates to the interval $[-1, 1]$ and used the first nine Chebyshev polynomials of the first kind.

For all methods but LDA, multidimensional $\mathbf{x}_i$ were projected onto their principal components prior to classification and thereafter rescaled to variance one. This operation amounts to multiplying the $\mathbf{x}_i$ with a whitening matrix, and the $\mathbf{y}_i$ were multiplied with the same matrix.

For cross-validation, $K = 5$ folds were used. The max-rule consisted in trying several classification methods and selecting the one giving the largest classification accuracy. We used L_1 and L_2 regularized polynomial logistic regression and SVM classification with the penalties $C = 0.1, 1, 10$, as well as LDA and QDA. When LDA was not applicable (as for the moving average model), it was excluded from the pool of classification methods used for the max-rule.

2.2 Models used for continuous, binary, count, and time series data

We tested our inference method on several well-known distributions. This section details the models and lists the parameters used to generate the data, as well as the priors employed for Bayesian inference and the corresponding posterior distributions. The posterior distributions served as reference against which we compared the distributions produced by classifier ABC.

The sample average of n data points $(x_1, \ldots, x_n)$ will be denoted by $\bar{x}$, and the sample variance by s_n^2,

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i, \qquad s_n^2 = \frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2. \tag{S14}$$

2.2.1 Continuous data

We considered inference for a univariate Gaussian with unknown mean and known variance, and inference of both mean and variance.

Gaussian with unknown mean. The data were sampled from a univariate Gaussian with mean $\mu^o = 1$ and variance $v^o = 1$. Inference was performed on the mean μ. In the Bayesian setting, the prior distribution of μ was Gaussian,

$$\mu \sim \mathcal{N}(\mu_0, v_0), \qquad p(\mu|\mu_0, v_0) = \frac{1}{\sqrt{2\pi v_0}} \exp\left(-\frac{(\mu-\mu_0)^2}{2v_0}\right), \tag{S15}$$

with mean $\mu_0 = 3$ and variance $v_0 = 1$. For Gaussian data with known variance v^o and a Gaussian prior on the mean μ, the posterior distribution of μ is Gaussian with mean μ_n and variance v_n,

$$\mu|\mathbf{X} \sim \mathcal{N}(\mu_n, v_n), \qquad \mu_n = \left(\frac{\mu_0}{v_0} + \frac{n\bar{x}}{v^o}\right) v_n, \qquad v_n = \left(\frac{1}{v_0} + \frac{n}{v^o}\right)^{-1}, \tag{S16}$$

see, for example, (Gelman et al. 2003, Chapter 2).

Gaussian with unknown mean and variance. The Gaussian data were generated with mean $\mu^o = 3$ and variance $v^o = 4$. Both mean μ and variance v were considered unknown. In the Bayesian setting, the prior distribution was normal-inverse-gamma,

$$\mu|v \sim \mathcal{N}\left(\mu_0, \frac{v}{\lambda_0}\right), \qquad v \sim \mathcal{G}^{-1}(\alpha_0, \beta_0), \qquad p(v|\alpha_0, \beta_0) = \frac{\beta_0^{\alpha_0}}{\Gamma(\alpha_0)} v^{-\alpha_0-1} \exp\left(-\frac{\beta_0}{v}\right), \tag{S17}$$

where α_0 and β_0 are the shape and scale parameters, respectively, and $\Gamma(.)$ is the gamma function, $\Gamma(t) = \int_0^\infty u^{t-1}\exp(-u)\mathrm{d}u$. The parameter values $\mu_0 = 0, \lambda_0 = 1, \alpha_0 = 3, \beta_0 = 0.5$ were used. This gives a prior variance with mean and standard deviation 0.25. The posterior is normal-inverse-gamma with updated parameters $\mu_n, \lambda_n, \alpha_n, \beta_n$,

$$\mu|v, \mathbf{X} \sim \mathcal{N}\left(\mu_n, \frac{v}{\lambda_n}\right), \qquad \mu_n = \frac{\lambda_0\mu_0 + n\bar{x}}{\lambda_0 + n}, \qquad \lambda_n = \lambda_0 + n, \tag{S18}$$

$$v|\mathbf{X} \sim \mathcal{G}^{-1}(\alpha_n, \beta_n), \qquad \alpha_n = \alpha_0 + \frac{n}{2}, \qquad \beta_n = \beta_0 + \frac{n}{2}s_n^2 + \frac{n}{2}\frac{\lambda_0}{\lambda_0 + n}(\bar{x} - \mu_0)^2, \tag{S19}$$

see, for example, (Gelman et al. 2003, Chapter 3).

2.2.2 Binary data

The data were a random sample from a Bernoulli distribution with success probability (mean) $\mu^o = 0.2$. The prior on the mean μ was a beta distribution with parameters $\alpha_0 = \beta_0 = 2$,

$$\mu \sim \mathrm{Beta}(\alpha_0, \beta_0), \qquad p(\mu|\alpha_0, \beta_0) = \frac{\Gamma(\alpha_0 + \beta_0)}{\Gamma(\alpha_0)\Gamma(\beta_0)} \mu^{\alpha_0 - 1}(1-\mu)^{\beta_0 - 1}, \tag{S20}$$

which has mean 0.5 and standard deviation 0.22. The posterior is beta with parameters α_n, β_n,

$$\mu|\mathbf{X} \sim \mathrm{Beta}(\alpha_n, \beta_n), \qquad \alpha_n = \alpha_0 + n\bar{x}, \qquad \beta_n = \beta + n(1-\bar{x}), \tag{S21}$$

see, for example, (Gelman et al. 2003, Chapter 2).

2.2.3 Count data

The data were a random sample from a Poisson distribution with mean $\lambda^o = 10$. The prior on the mean λ was a gamma distribution with shape parameter $\alpha_0 = 3$ and rate parameter $\beta_0 = 1/2$,

$$\lambda \sim \mathcal{G}(\alpha_0, \beta_0), \qquad p(\lambda|\alpha_0, \beta_0) = \frac{\beta_0^{\alpha_0}}{\Gamma(\alpha_0)} \lambda^{\alpha_0 - 1} \exp\left(-\beta_0\lambda\right). \tag{S22}$$

The prior distribution has mean 6, mode 4, and standard deviation 3.46. The posterior distribution is gamma with parameters α_n, β_n,

$$\lambda|\mathbf{X} \sim \mathcal{G}(\alpha_n, \beta_n), \qquad \alpha_n = \alpha_0 + n\bar{x}, \qquad \beta_n = \beta_0 + n, \tag{S23}$$

see, for example, (Gelman et al. 2003, Chapter 2).

2.2.4 Time series

We considered a moving average and an ARCH(1) model.

Moving average model. The time series is determined by the update equation

$$x_t = \epsilon_t + \theta\epsilon_{t-1}, \quad t = 1, \ldots, T, \tag{S24}$$

where the $\epsilon_t, t = 0, \ldots, T$ are independent standard normal random variables, and ϵ_0 is unobserved. The observed data were generated with $\theta^o = 0.3$. The $\mathbf{x}_i$ for classification consisted of 2 consecutive time points (x_t, x_{t+1}).

For the derivation of the posterior distribution, it is helpful to write the update equation in matrix form. Let $\mathbf{x}_{0:T} = (x_0, \ldots, x_T)$ and $\boldsymbol{\epsilon} = (\epsilon_0, \ldots, \epsilon_T)$ be two column vectors of length $T+1$. The update equation does not specify the value of x_0. We thus set $x_0 = \epsilon_0$. Equation (S24) can then be written as

$$\mathbf{x}_{0:T} = \mathbf{B}\boldsymbol{\epsilon}, \qquad \mathbf{B} = \begin{pmatrix} 1 & 0 & & & \\ \theta & 1 & 0 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 0 \\ & & & \theta & 1 \end{pmatrix}. \tag{S25}$$

It follows that $\mathbf{x}_{0:T}$ is zero mean Gaussian with covariance matrix $\mathbf{B}\mathbf{B}^\top$. Since $\mathbf{x}_{0:T}$ has a Gaussian distribution, we can analytically integrate out the unobserved x_0. The resulting vector $\mathbf{x}_{1:T}$ is zero mean Gaussian with tridiagonal covariance matrix $\mathbf{C}$,

$$\mathbf{C} = \begin{pmatrix} 1+\theta^2 & \theta & & & \\ \theta & 1+\theta^2 & \theta & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \theta \\ & & & \theta & 1+\theta^2 \end{pmatrix}. \tag{S26}$$

We denote the distribution of $\mathbf{x}_{1:T}$ by $p(\mathbf{x}_{1:T}|\theta)$. A uniform prior on $(-1, 1)$ was assumed for θ. The posterior probability density function of θ given $\mathbf{x}_{1:T}$ is thus $p(\theta|\mathbf{x}_{1:T})$,

$$p(\theta|\mathbf{x}_{1:T}) = \frac{p(\mathbf{x}_{1:T}|\theta)}{\int_{-1}^{1} p(\mathbf{x}_{1:T}|\theta)\mathrm{d}\theta}, \qquad \theta \in (-1, 1). \tag{S27}$$

The normalizing denominator can be computed using numerical integration. Numerical integration can also be used to compute the posterior mean and variance. We used matlab's `integral.m`.

ARCH(1) model. The model used was

$$x_t = \theta_1 x_{t-1} + \epsilon_t, \qquad \epsilon_t = \xi_t\sqrt{0.2 + \theta_2\epsilon_{t-1}^2}, \qquad t = 1, \ldots, T, \qquad x_0 = 0, \tag{S28}$$

where the ξ_t and ϵ_0 are independent standard normal random variables. We call θ_1 the mean process coefficient and θ_2 the variance process coefficient. The observed data consist of the x_t and we generated them with $(\theta_1^o, \theta_2^o) = (0.3, 0.7)$. The $\mathbf{x}_i$ used for classification consisted of 5 consecutive time points.

For the derivation of the posterior distribution, we introduce the column vectors $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_T)$ and $\mathbf{x}_{1:T} = (x_1, \ldots, x_T)$ which are related by a linear transformation,

$$\boldsymbol{\epsilon} = \mathbf{Q}\mathbf{x}_{1:T}, \qquad \mathbf{Q} = \begin{pmatrix} 1 & 0 & & & \\ -\theta_1 & 1 & 0 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 0 \\ & & & -\theta_1 & 1 \end{pmatrix}. \tag{S29}$$

Note that the band-diagonal matrix $\mathbf{Q}$ depends on θ_1. The determinant of $\mathbf{Q}$ is one so that

$$p_{\mathbf{x}}(\mathbf{x}_{1:T}|\theta_1, \theta_2) = p_\epsilon(\mathbf{Q}\mathbf{x}_{1:T}|\theta_1, \theta_2). \tag{S30}$$

The assumption on the ξ_t implies that $\epsilon_t|\epsilon_{t-1}$ is Gaussian with variance $0.2 + \theta_2\epsilon_{t-1}^2$. We thus have

$$p_\epsilon(\boldsymbol{\epsilon}|\theta_1, \theta_2) = p_1(\epsilon_1|\theta_1, \theta_2)\prod_{t=2}^{T}\frac{1}{\sqrt{2\pi(0.2 + \theta_2\epsilon_{t-1}^2)}}\exp\left(-\frac{\epsilon_t^2}{2(0.2 + \theta_2\epsilon_{t-1}^2)}\right), \tag{S31}$$

where p_1 is the pdf of ϵ_1. Since ϵ_0 is a latent variable following a standard normal distribution, p_1 is defined via an integral,

$$p_1(\epsilon_1|\theta_1, \theta_2) = \int \frac{1}{\sqrt{2\pi(0.2 + \theta_2\epsilon_0^2)}}\exp\left(-\frac{\epsilon_1^2}{2(0.2 + \theta_2\epsilon_0^2)}\right)\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{\epsilon_0^2}{2}\right)\mathrm{d}\epsilon_0. \tag{S32}$$

We used numerical integration, matlab's `integral.m`, to evaluate it. The prior distribution of (θ_1, θ_2) was the uniform distribution on the rectangle $(-1, 1) \times (0, 1)$. The posterior pdf $p(\theta_1, \theta_2|\mathbf{x}_{1:T})$ is

$$p(\theta_1, \theta_2|\mathbf{x}_{1:T}) = \frac{p_\epsilon(\mathbf{Q}\mathbf{x}_{1:T}|\theta_1, \theta_2)}{\int_{-1}^{1}\int_{0}^{1} p_\epsilon(\mathbf{Q}\mathbf{x}_{1:T}|\theta_1, \theta_2)\mathrm{d}\theta_1\mathrm{d}\theta_2}, \qquad (\theta_1, \theta_2) \in (-1, 1) \times (0, 1). \tag{S33}$$

The normalizing denominator, the posterior means and variances were computed with matlab's `integral2.m`.

2.3 ABC algorithm

There are several algorithms for approximate Bayesian computation (ABC, for an overview, see, for example, Marin et al. 2012). For the results in the paper, we used a population Monte Carlo sampler, also known as sequential Monte Carlo ABC algorithm, with a Gaussian kernel (Marin et al. 2012, Algorithm 4), (Sisson et al. 2007; Beaumont et al. 2009; Toni et al. 2009). In brief, the algorithm starts with samples from the prior distribution and then produces sets (generations) of weighted independent samples where the samples from any given generation are the starting point to get the samples of the next generation. The empirical pdfs, scatter plots, and sample moments reported in the paper all take the weights into account.

In some ABC implementations, the acceptance thresholds are the empirical quantiles of the discrepancies of the accepted parameters; in others, a schedule is pre-defined. The pre-defined schedule depends on the scale of the discrepancy measure which is often unknown. Using quantiles avoids this problem, but if the quantile is set too low, too few samples will be accepted which results in a slow algorithm. For J_n, the scale is known. We took advantage of this and used a hybrid approach to choose the thresholds: The threshold for a generation was the maximum of the value given by a pre-defined schedule and the value given by the 0.1 quantile of the J_n of the accepted parameters from the previous generation. With t denoting the ABC generation, the schedule was $0.75/(1 + 0.45 \log t)$, which gives a value of 0.5 at $t = 3$.

Unlike a purely quantile-based approach, the hybrid approach avoids sudden jumps to small thresholds. We can thereby obtain posteriors for intermediate thresholds. These are faster to obtain but still informative. The final posteriors from both approaches are, however, very similar, as shown in Supplementary Figure 1.

2.4 Application to infectious disease epidemiology

We here summarize the data, model and inference method used by Numminen et al. (2013).

2.4.1 Data and model

The data consisted of the colonization state of the attendees of $K = 29$ day care centers at certain points of time T_k (cross-sectional data). For each day care center, only a subset of size N_k of all attendees was sampled. The colonization state of individual i in a day care center was represented by binary variables $I_{is}^t, s = 1, \ldots, S$, where $I_{is}^t = 1$ means that attendee i is infected with strain s at time t. The observed data $\mathbf{X}$ consisted thus of a set of 29 binary matrices of size $N_k \times S$ formed by the $I_{is}^{T_k}$, $i = 1, \ldots, N_k, s = 1, \ldots, S$.

The data were modeled using a continuous-time Markov chain for the transmission dynamics within a day care center, and an observation model (Numminen et al. 2013). The day care centers were assumed independent.

We first review the model for the transmission dynamics within a day care center: Starting with zero infected individuals, $I_{is}^0 = 0$ for all i and s, the states were assumed to evolve in a stochastic way,

$$\mathrm{P}(I_{is}^{t+h} = 0|I_{is}^t = 1) = h + o(h), \tag{S34}$$

$$\mathrm{P}(I_{is}^{t+h} = 1|I_{is'}^t = 0\ \forall s') = R_s(t)h + o(h), \tag{S35}$$

$$\mathrm{P}(I_{is}^{t+h} = 1|I_{is}^t = 0,\ \exists s' : I_{is'}^t = 1) = \theta R_s(t)h + o(h), \tag{S36}$$

where h is a small time interval and $o(h)$ a negligible remainder term of smaller order satisfying $\lim_{h\to 0} o(h)/h = 0$. The first equation describes the probability to clear strain s, the second equation the probability to be infected by it when previously not infected with any strain, and the last equation, the probability to be infected by it when previously infected with another strain s'. The rate of infection with strain s at time t is denoted by $R_s(t)$, and $\theta \in (0, 1)$ is an unknown co-infection parameter. For $\theta = 0$, the probability for a co-infection is zero. The rate $R_s(t)$ was modeled as

$$R_s(t) = \beta E_s(t) + \Lambda P_s, \qquad E_s(t) = \sum_{j=1}^{N} \frac{1}{N-1}\frac{I_{js}^t}{n_j(t)}, \qquad n_j(t) = \sum_{s'=1}^{S} I_{js'}^t, \tag{S37}$$

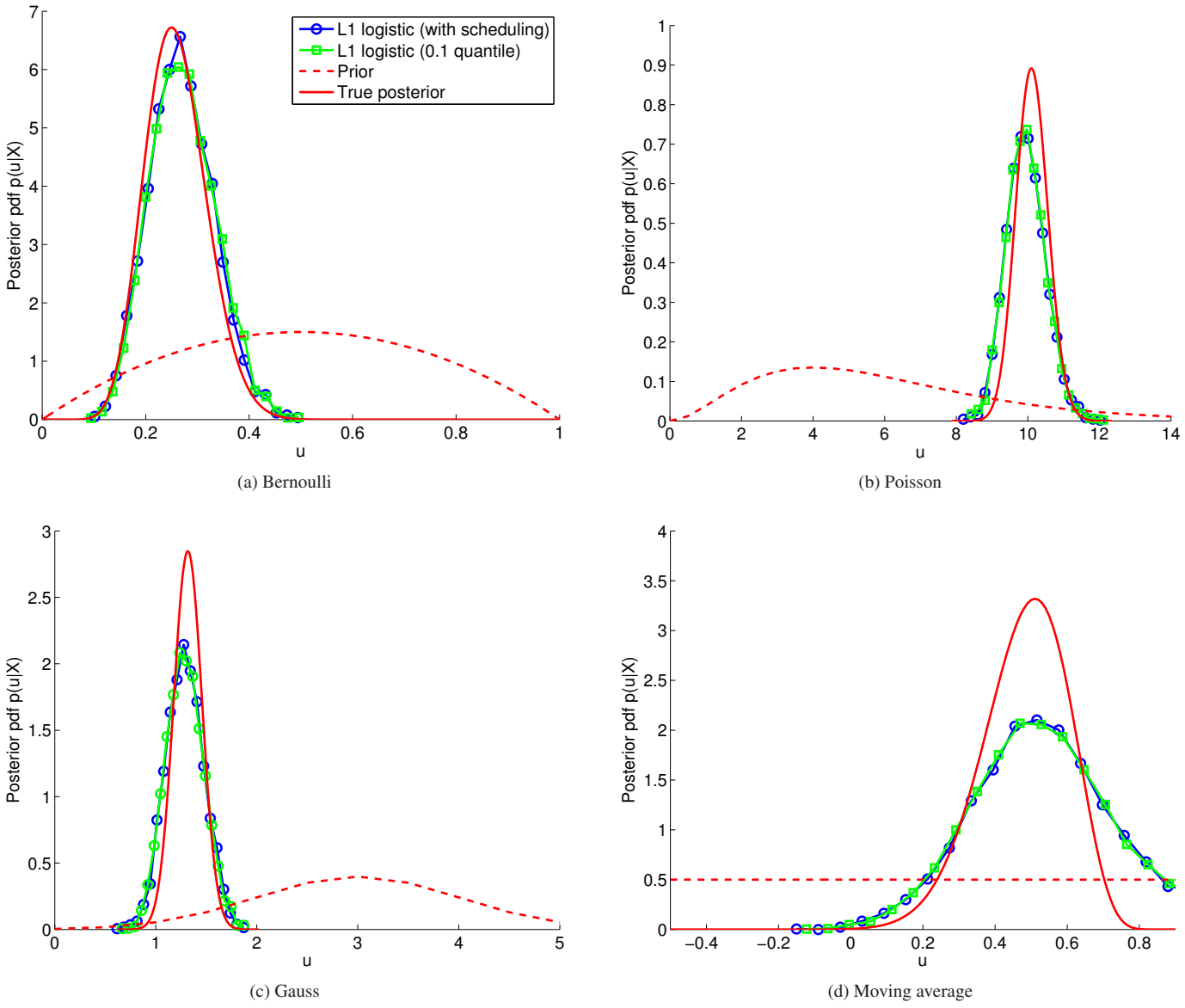
where N is the number of children attending the day care center, and S the total number of strains in circulation. Λ and β are two unknown rate parameters which scale the (static) probability P_s for an infection happening outside the day care center and the (dynamic) probability $E_s(t)$ for an infection from within, respectively. The probability P_s and the number of strains S were determined by an analysis of the overall distribution of the strains in the cross-sectional data (yielding $S = 33$, for P_s, see the original paper by Numminen et al. (2013)). The expression for $E_s(t)$ was derived by assuming that contact with another attendee j is uniformly at random (the probability for a contact is then $1/(N-1)$), and assuming an equal probability for a transmission of any of the strains attendee j is carrying (with $n_j(t)$ being the total number of strains carried by j, the probability for a transmission of strain s is $I_{js}^t/n_j(t)$).

The observation model was random sampling of N_k individuals without replacement from the $N = 53$ individuals attending a day care center. The value of N was set to the average number of attendees of the 29 day care centers. A stationarity assumption was made so that the exact value of T_k was not of importance as long as it is sufficiently large so that the system is in its stationary regime.

The model has three unknown parameters: the internal infection parameter β, the external infection parameter Λ, and the co-infection parameter θ.

2.4.2 Expert statistics and distance used in ABC

Numminen et al. (2013) used a sequential Monte Carlo implementation of approximate Bayesian computation (ABC) to obtain posterior distributions of the three parameters $\boldsymbol{\theta} = (\beta, \Lambda, \theta)$ assuming uniform priors for $\beta \in (0, 11)$, $\Lambda \in (0, 2)$, and $\theta \in (0, 1)$. This involved computing the distance between the observed data $\mathbf{X}$ and data $\mathbf{Y}_{\boldsymbol{\theta}}$ simulated with parameter $\boldsymbol{\theta}$. Both observed and simulated data consisted of 29 binary matrices of different sizes, corresponding to the observed states of the different day care centers. Each day care center was summarized using four statistics: (a) the diversity of the strains present, (b) the number of different strains present, (c) the proportion of infected individuals, (d) the proportion of individuals with more than one strain. Each data set was then represented by the empirical cumulative distribution functions (cdfs) of the four summary statistics, computed over the 29 day care centers which form the data set. The L_1 distance between the empirical cdfs of $\mathbf{X}$ and $\mathbf{Y}_{\boldsymbol{\theta}}$ were computed for each summary statistic, and $\boldsymbol{\theta}$ was accepted if each of the four L_1 distances was below a certain threshold. The four thresholds were adapted during the algorithm (Numminen et al. 2013).



(a) Bernoulli

(b) Poisson

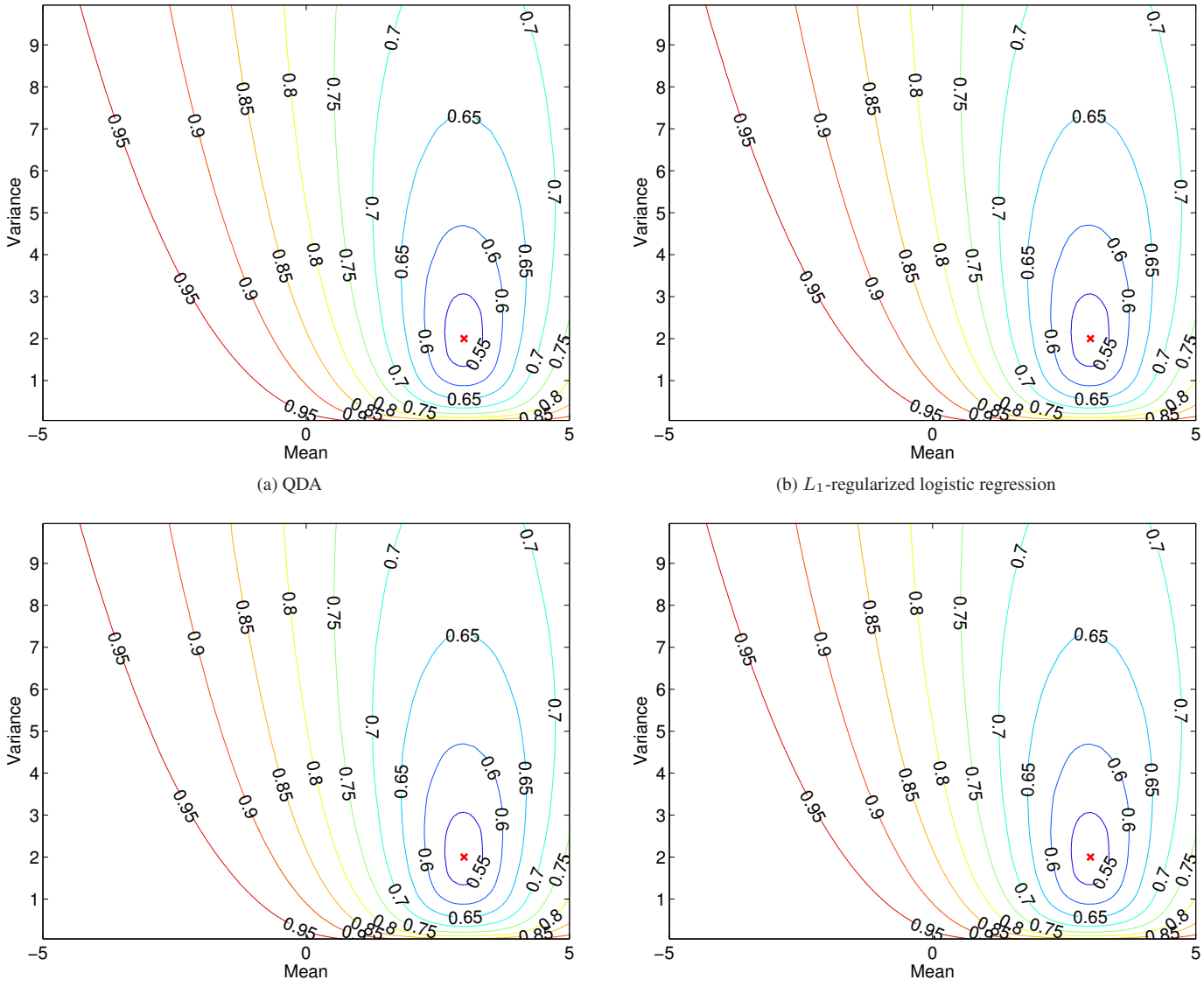
(c) Gauss

(d) Moving average

Supplementary Figure 1: Assessment of the hybrid approach to choose the acceptance thresholds in classifier ABC with a sequential Monte Carlo algorithm. The final posterior pdfs for the hybrid approach (blue, circles) and a purely quantile-based approach (green, squares) are very similar. The benefit of the hybrid approach is that it yields more quickly useful intermediate solutions. The results are for L_1-regularized polynomial logistic regression.

3 MEASURING DISCREPANCY VIA CLASSIFICATION

In Figure 2 in the main text, chance-level discriminability was attained at a point close to the parameter $\boldsymbol{\theta}^o$ which was used to generate $\mathbf{X}$. We provide here two more such examples: Supplementary Figure 2 shows the results for a Gaussian distribution with unknown mean and variance, and Supplementary Figure 3 the results for the autoregressive conditional heteroskedasticity (ARCH) time series model in Equation (S28) with unknown mean and variance process coefficients. Parameter $\boldsymbol{\theta}^o$ is marked with a red cross.



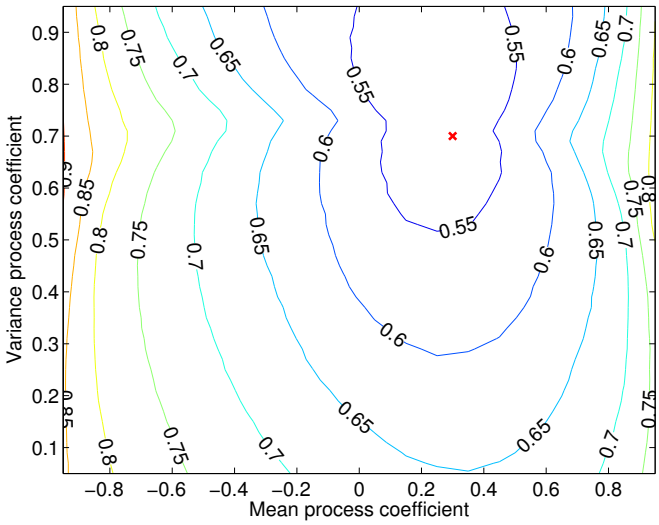
(a) QDA

(b) L_1-regularized logistic regression

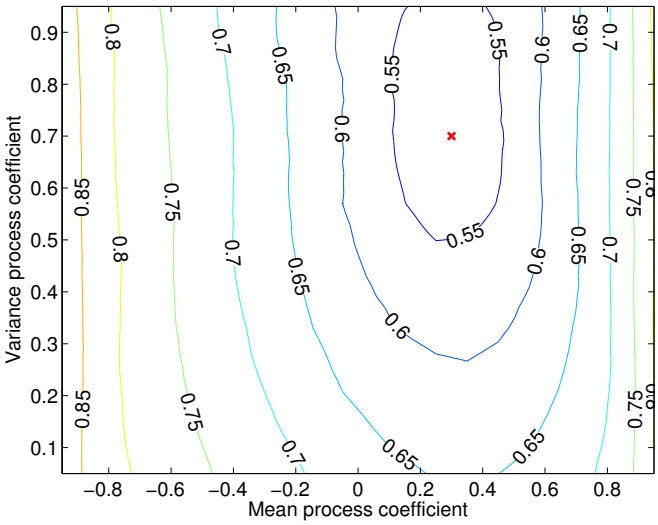
(c) L_1-regularized SVM

(d) Max-Rule

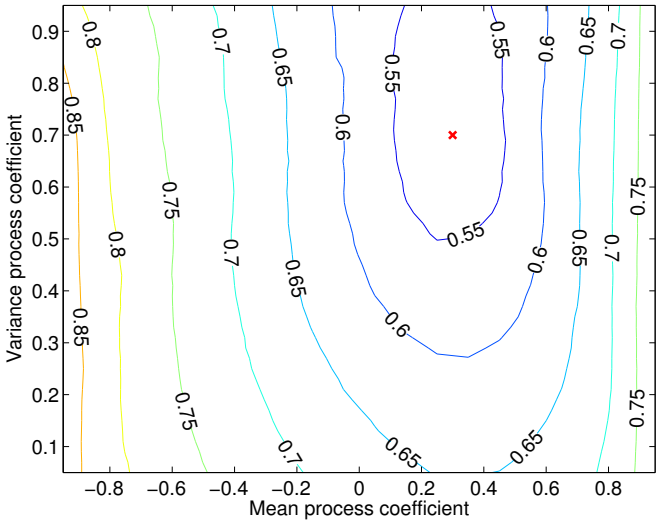
Supplementary Figure 2: Gaussian with unknown mean and variance. The contour plots show J_n as a function of the two parameters for large sample sizes ($n = 100{,}000$). The different panels depict results for different classification methods. All obtain their minimal classification accuracy, chance-level discriminability 0.5, close to $\boldsymbol{\theta}^o$.



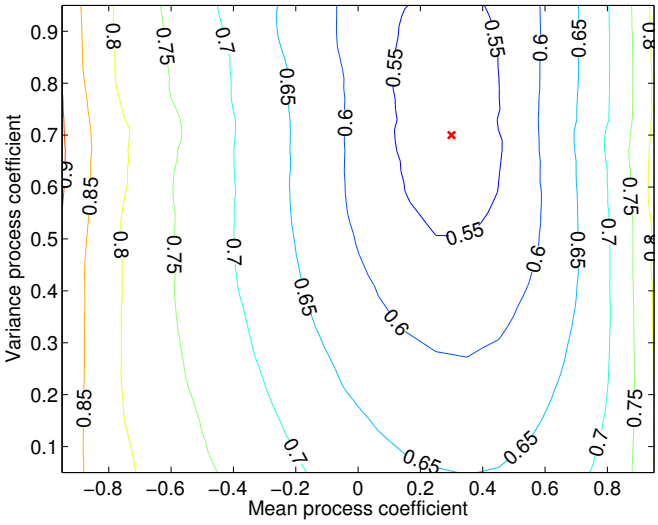
(a) QDA



(b) L_1-regularized logistic regression



(c) L_1-regularized SVM

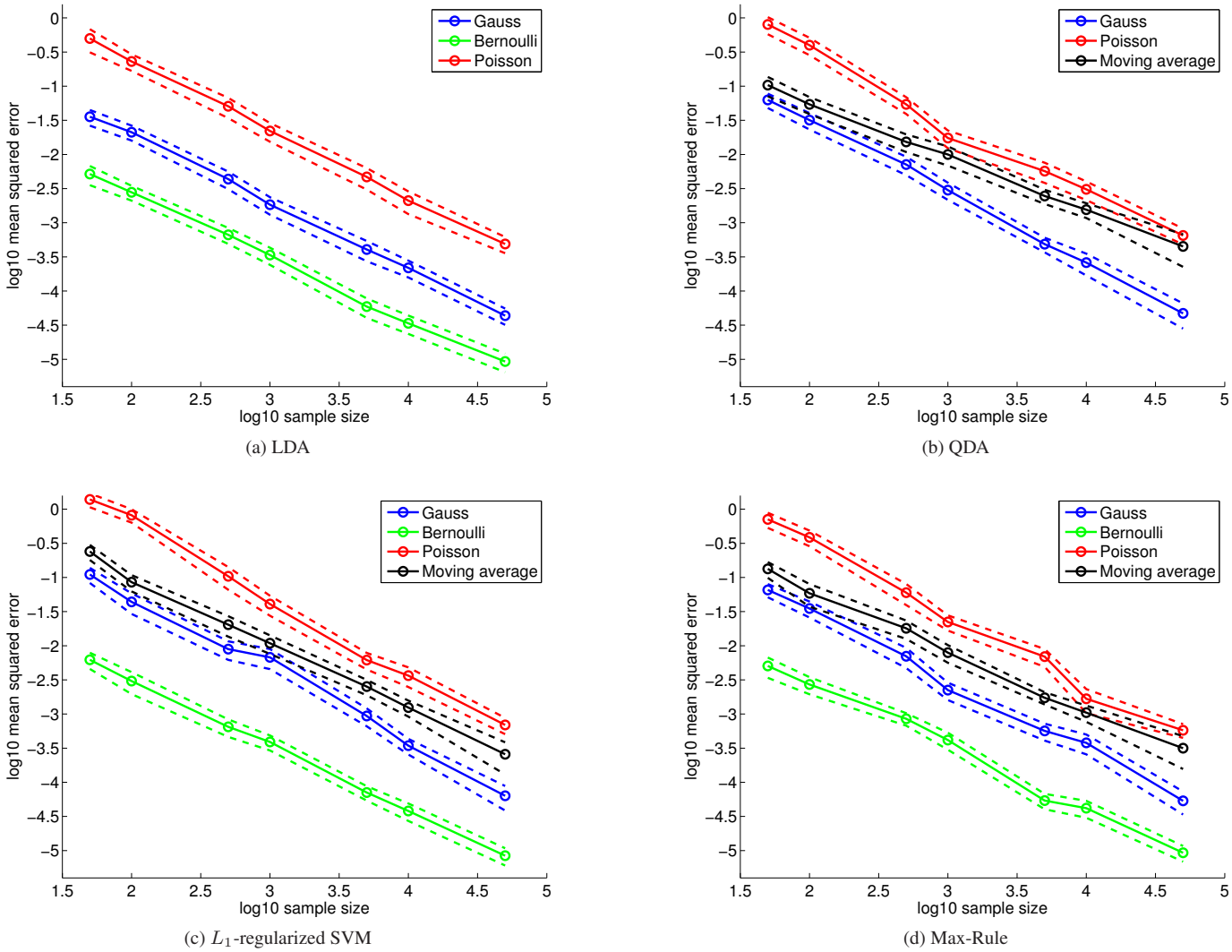


(d) Max-Rule

Supplementary Figure 3: ARCH(1) model in Equation (S28) with unknown mean and variance process coefficients θ_1 and θ_2. The results are for $n = 10{,}000$ and visualized as in Supplementary Figure 2.

4 POINT ESTIMATION VIA CLASSIFICATION

In Figure 3 in the main text, we plotted the mean squared estimation error $\mathrm{E}[||\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}^o||^2]$ for the examples in Figure 2 against the sample size n for L_1-regularized logistic regression. Supplementary Figure 4 shows the corresponding results for linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), L_1-regularized polynomial support vector machine (SVM) classification, and the max-rule. As for the results in the main text, the decay is linear on the log-log scale which suggests convergence in quadratic mean, hence convergence in probability, and thus consistency of $\hat{\boldsymbol{\theta}}_n$.



(a) LDA

(b) QDA

(c) L_1-regularized SVM

(d) Max-Rule

Supplementary Figure 4: The mean squared estimation error for the examples in Figure 2 in the main text as a function of the sample size n (solid lines, circles). The mean was computed as an average over 100 outcomes. The dashed lines depict the mean $\pm$ 2 standard errors. For QDA, the Bernoulli case is not reported because, sometimes, data with degenerate covariance matrices were generated, which the standard QDA algorithm used was not able to handle. For LDA, the moving average case was omitted since LDA cannot approximate its Bayes classification rule. We discuss this point in the main text. The linear trend on the log-log scale suggests convergence in quadratic mean, and hence consistency of the estimator $\hat{\boldsymbol{\theta}}_n$.

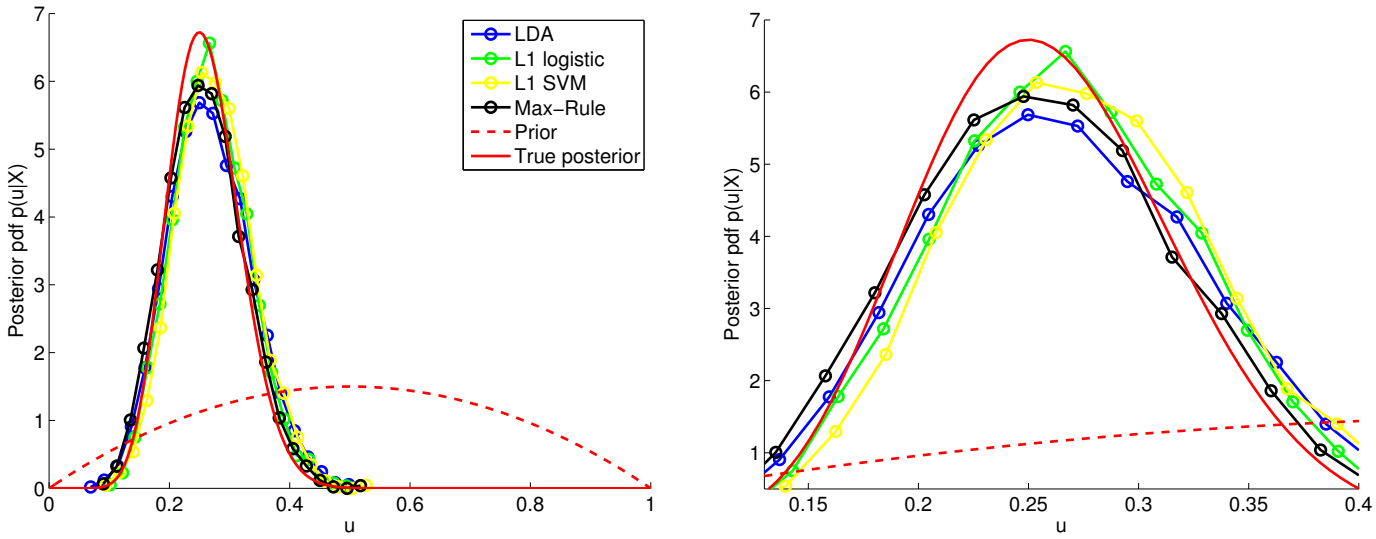
5 Approximate Bayesian computation via classification

This section contains further results for classifier ABC on data with known properties.

5.1 The inferred posterior distributions for all classification methods used

We report the posterior distributions for all classification methods used in the paper in Supplementary Figure 5 to Supplementary Figure 10. The results are organized according to the modality of the data.

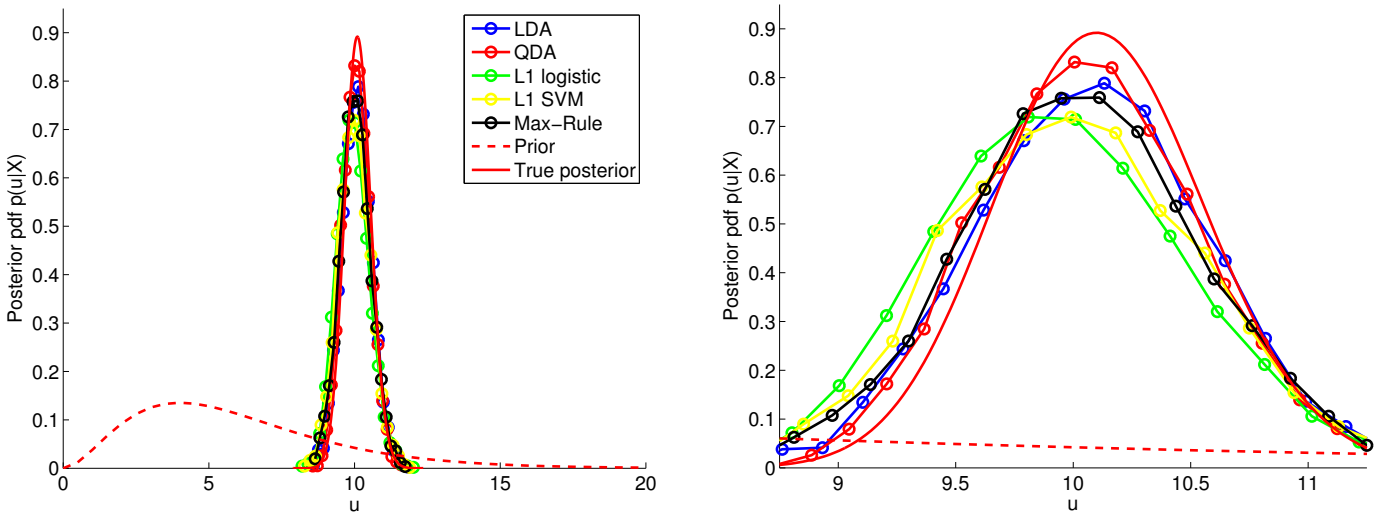
The results are for $n = 50$ and 10,000 ABC samples with a sequential Monte Carlo implementation of ABC. For the univariate cases, empirical pdfs of the ABC samples are shown together with the reference posterior pdf (red solid) and the prior pdf used (red dashed). For the bivariate cases, the ABC samples are shown as a scatter plot and the reference posterior is visualized using contour plots (red solid line). The priors are either shown as contour plots (with red dashed lines) or, if uniform, by hatching their domain.



(a) Prior and posterior

(b) Zoom

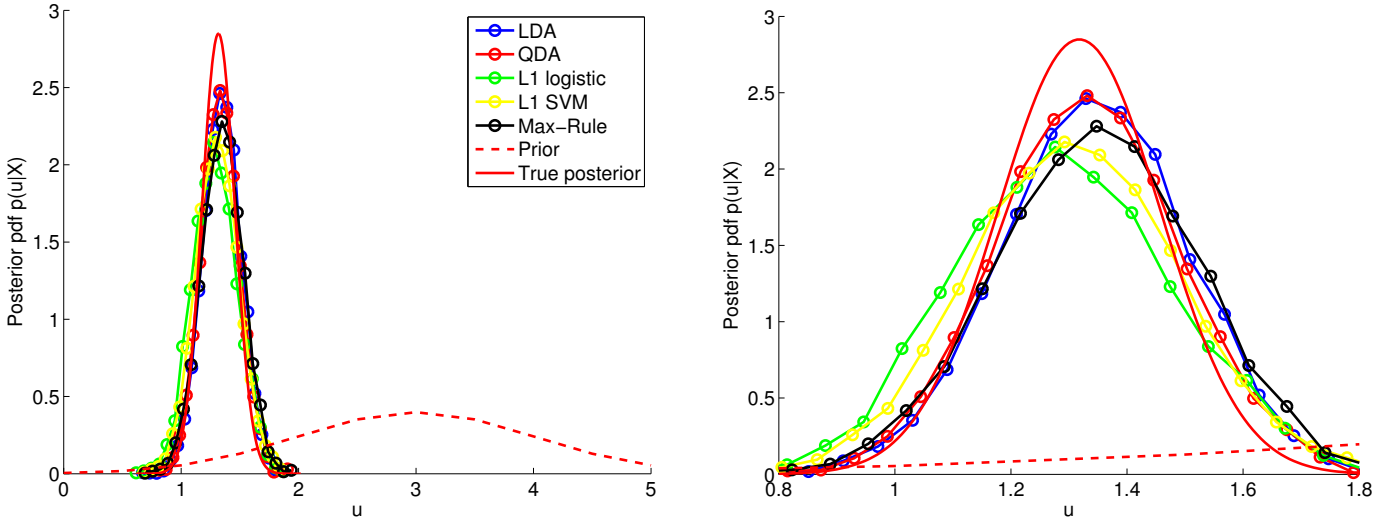
Supplementary Figure 5: Binary data: Inferred posterior distribution of the success probability of a Bernoulli random variable.



(a) Prior and posterior

(b) Zoom

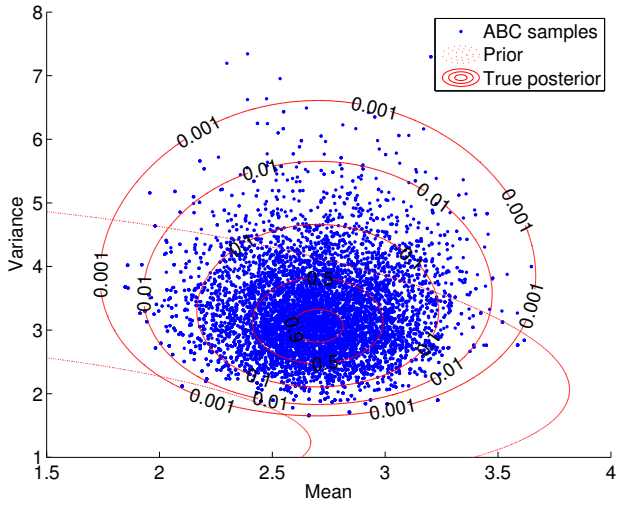
Supplementary Figure 6: Count data: Inferred posterior distribution of the mean of a Poisson random variable.



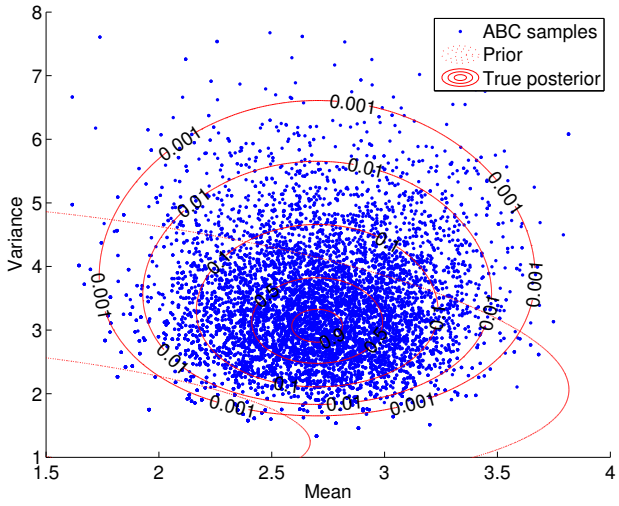
(a) Prior and posterior

(b) Zoom

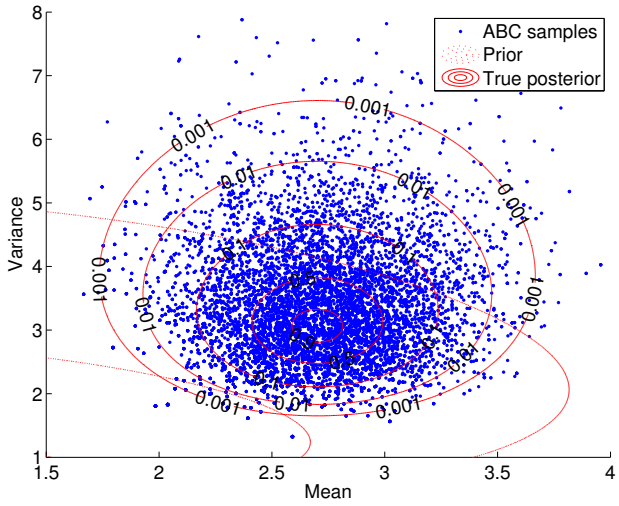
Supplementary Figure 7: Continuous data: Inferred posterior distribution of the mean of a Gaussian random variable with known variance.



(a) QDA

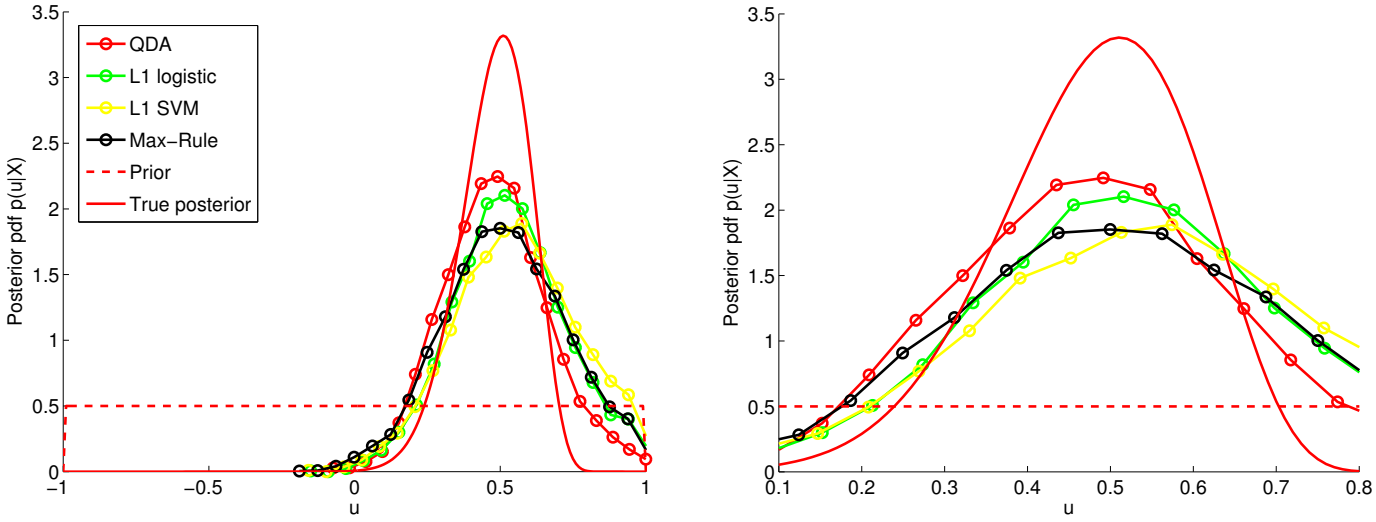


(b) L_1-regularized logistic regression



(c) L_1-regularized SVM

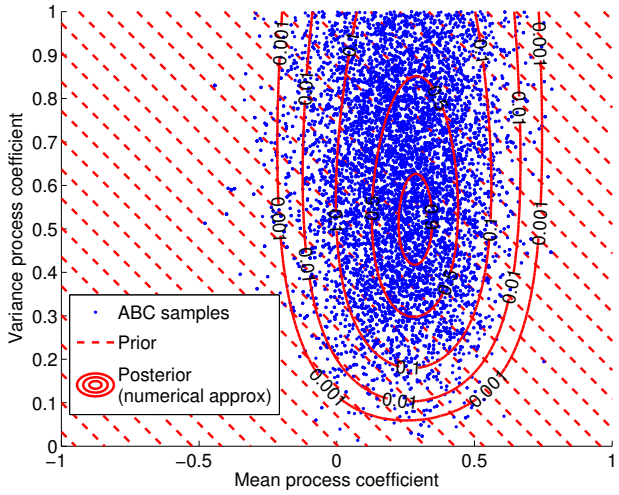
Supplementary Figure 8: Continuous data: Inferred posterior distribution of the mean and variance of a Gaussian random variable.



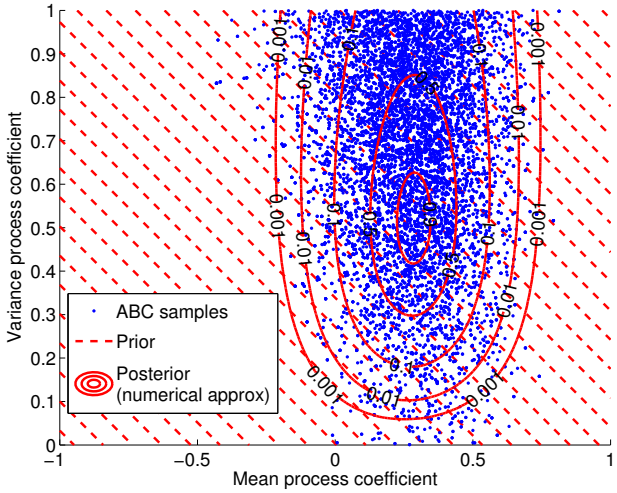
(a) Prior and posterior

(b) Zoom

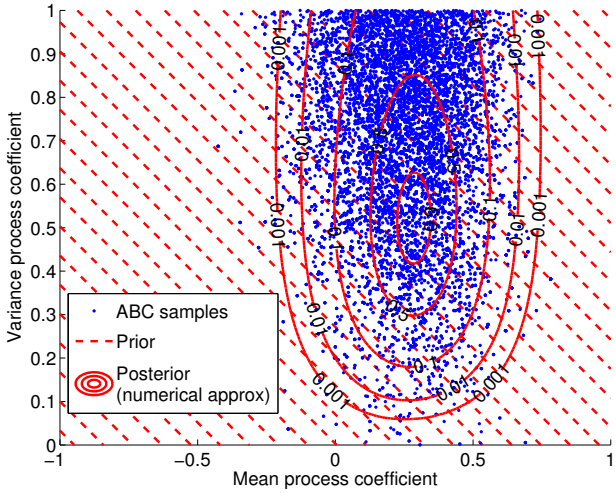
Supplementary Figure 9: Time series: Inferred posterior distribution of the lag coefficient of a zero mean moving average model of order one.



(a) QDA



(b) L_1-regularized logistic regression



(c) L_1-regularized SVM

Supplementary Figure 10: Time series: Inferred posterior distribution of the mean and variance process coefficients of a ARCH(1) model.

5.2 Movies showing the evolution of the inferred posteriors

The sequential Monte Carlo algorithm which we used together with classifier ABC is iteratively morphing a prior distribution into a posterior distribution. Table 1 contains links to movies which show this process.

Data	LDA	QDA	Logi regr	SVM	Max-Rule
Binary (Bernoulli)	avi mp4		avi mp4	avi mp4	avi mp4
Count (Poisson)	avi mp4	avi mp4	avi mp4	avi mp4	avi mp4
Continuous (Gauss, mean)	avi mp4	avi mp4	avi mp4	avi mp4	avi mp4
Continuous (Gauss, mean & var)		avi mp4	avi mp4	avi mp4	avi mp4
Time series (moving average)		avi mp4	avi mp4	avi mp4	avi mp4
Time series (ARCH)		avi mp4	avi mp4	avi mp4	avi mp4

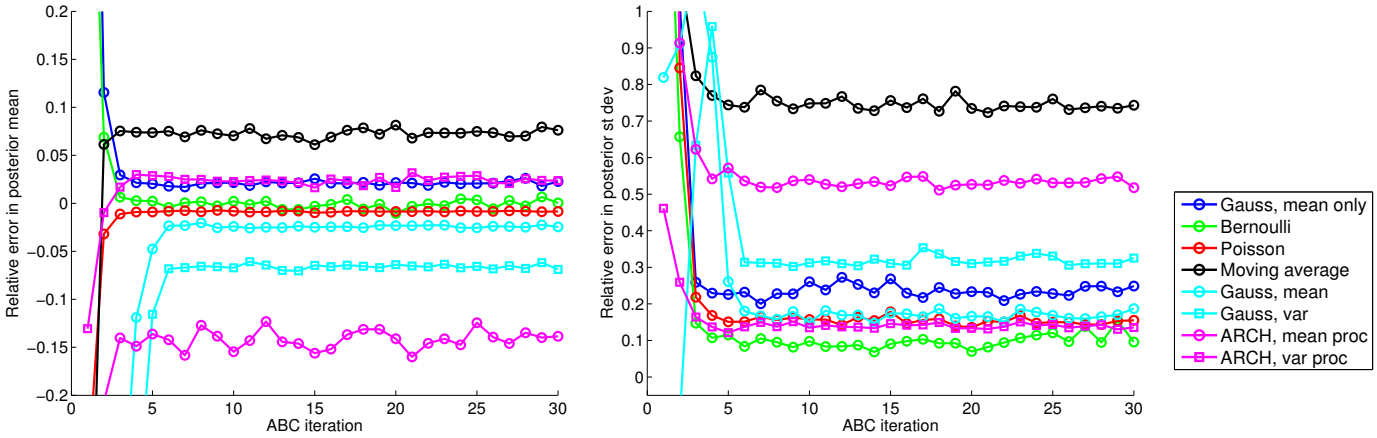
TABLE 1: Links to movies showing the inference process of classifier ABC with a sequential Monte Carlo algorithm. Online at https://drive.google.com/folderview?id=0B1HTVHlhiD81RTZ5VFNndG1RVXM&usp=drive_web

5.3 Relative errors in posterior means and standard deviations

As a quantitative analysis, we computed the relative error in the mean and the standard deviation of the inferred posterior distributions. The comparison is based on the mean and standard deviation of the true posterior if available, or the posterior obtained by deterministic numerical integration if not, see Supplementary Material 2.2.

Supplementary Figure 11 shows the relative error for the max-rule as a function of the iteration in the ABC algorithm. The error stabilizes within 4-5 iterations. For the examples with independent data points, the errors in the posterior mean are within 5% after stabilization. A larger error of 15% occurs for the time series data. The histograms and scatter plots show, however, that the corresponding ABC samples are still very reasonable.

While the relative error for the mean is both positive or negative, for the standard deviation, the error is positive only. This means that the inferred posteriors have a larger spread than the reference posteriors, that is, the posterior variance is overestimated. Further, the relative errors are generally larger for the standard deviations than for the means. This may not be too surprising though: Also in the framework of maximum likelihood estimation, the variance of the estimate of the variance is twice the variance of the estimate of the mean for standard normal random variables.



(a) Relative error in posterior mean

(b) Relative error in posterior standard deviation

Supplementary Figure 11: Quantitative analysis of the inferred posterior distributions. The curves show the relative error in the posterior mean and standard deviation for the Gauss, Bernoulli, Poisson, moving average, and ARCH examples. The results are for classification with the max-rule.

6 Application to infectious disease epidemiology

This section contains further results and analysis of our application to infectious disease epidemiology.

6.1 Preliminary investigation on simulated data

We investigated the applicability of LDA and the chosen features using $\mathbf{X}$ which was simulated from the model. The simulated $\mathbf{X}$ had the same size and structure as the actually observed data. Such preliminary investigations can always be done in the framework of ABC.

We first tested the applicability in the framework of classical inference. For that purpose, we computed $J_n(\boldsymbol{\theta})$ varying only two of the three parameters at a time. The third parameter was kept fixed at the value which was used to generate the data. In order to eliminate random effects, we used for all $\boldsymbol{\theta}$ the same random seed when simulating the $\mathbf{Y}_{\boldsymbol{\theta}}$. The random seeds for $\mathbf{X}$ and the $\mathbf{Y}_{\boldsymbol{\theta}}$ were different.

Supplementary Figure 12 shows the results for classification with randomly chosen subsets (top row) and without (bottom row). The diagrams on the top and bottom row are very similar, both have well-defined regions in the parameter space for which J_n is close to one half. But the features from the random subsets were helpful to discriminate between $\mathbf{X}$ and $\mathbf{Y}_{\boldsymbol{\theta}}$ and produced slightly more localized regions with small J_n.

We then ran sequential Monte Carlo ABC with J_n as distance measure (here, as usual in ABC, the $\mathbf{Y}_{\boldsymbol{\theta}}$ were generated with different random seeds). Supplementary Figure 13 visualizes the evolution of the inferred posterior distribution over four generations. We show the results for classifier ABC with random subsets (blue, circles) and without (red, squares). For reference, the results with the method of Numminen et al. (2013) using expert-knowledge are shown in black (point markers). The results for the fourth generation are shown separately in Supplementary Figure 14. The different solutions are qualitatively very similar, even though the tails of the posterior of β are heavier for classifier ABC than for the method of Numminen et al. (2013).

Numminen et al. (2013) showed posterior distributions for four generations. In both the results reported in Supplementary Figure 13 and the results by Numminen et al. (2013), the mean of the inferred posteriors seems to stabilize after four generations. The spread of the inferred posteriors, however, is still slightly shrinking. We thus ran the simulations for an additional fifth iteration. The results are shown in Supplementary Figure 15. With the fifth iteration, the posterior pdfs for classifier ABC with random projections became more concentrated and also more similar to the expert solution than the posteriors of classifier ABC without random projections. The smaller posterior variance is in line with the tighter J_n-diagrams in Supplementary Figure 12.

6.2 Evolution of inferred posterior distributions on real data

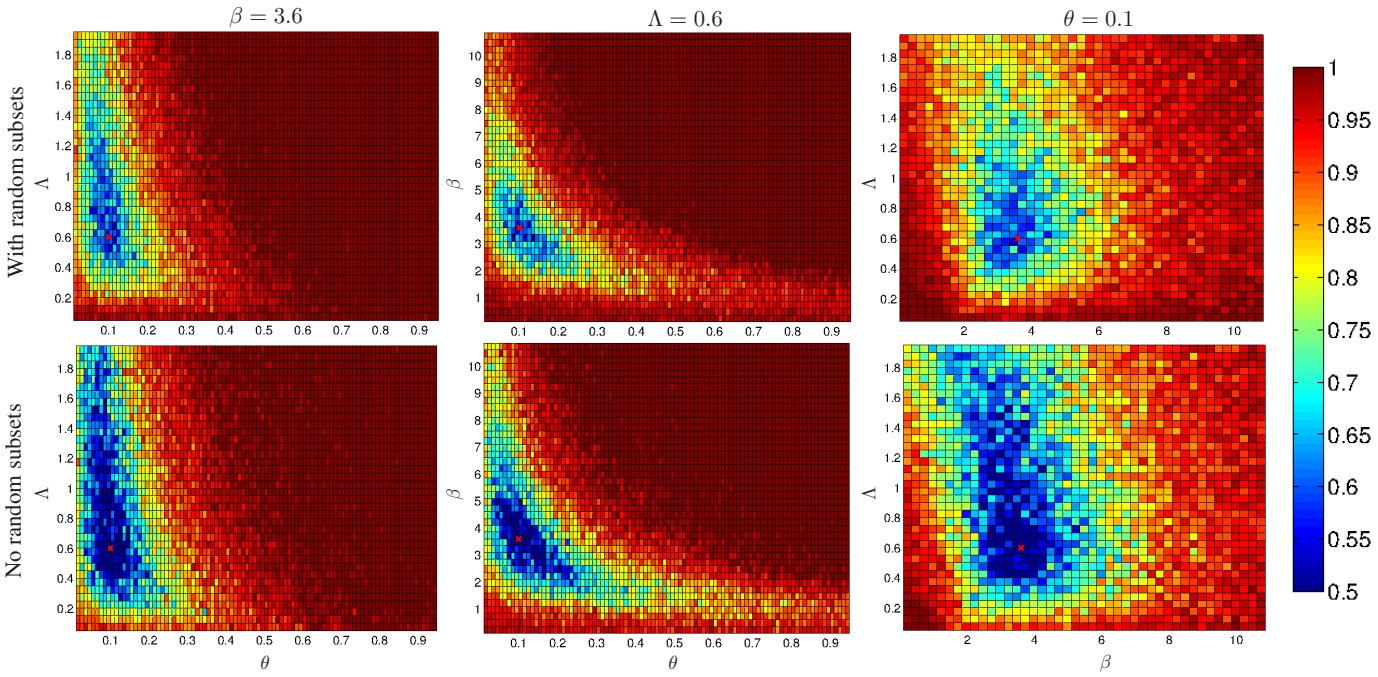
The evolution of the posterior pdfs during the ABC algorithm is shown in Supplementary Figure 16. Starting from uniform distributions, posterior distributions with well defined modes emerged. The results for the fourth generation are shown in more detail in Supplementary Figure 17. While the posteriors of Λ and θ are qualitatively similar for all three methods, the posterior of β has a smaller mode for classifier ABC with random subsets (blue, crosses) than for classifier ABC without random subsets (red, asterisks) or the expert solution (black, plus markers). This behavior persists in the fifth generation shown in Supplementary Figure 18. Compared to the fourth generation results, the posteriors for classifier ABC with random subsets (blue, crosses) and the expert solution (black, plus markers) became in the fifth generation more concentrated than the posterior for classifier ABC without random subsets (red, asterisks).

The results for real and simulated data are qualitatively similar. The main difference is the small shift in the posterior mode of β for classifier ABC with random subsets. This difference could be due to stochastic variation because we only worked with 1,000 ABC samples. It could, however, also be that the random features picked up some properties of the real data which the other methods are not sensitive to.

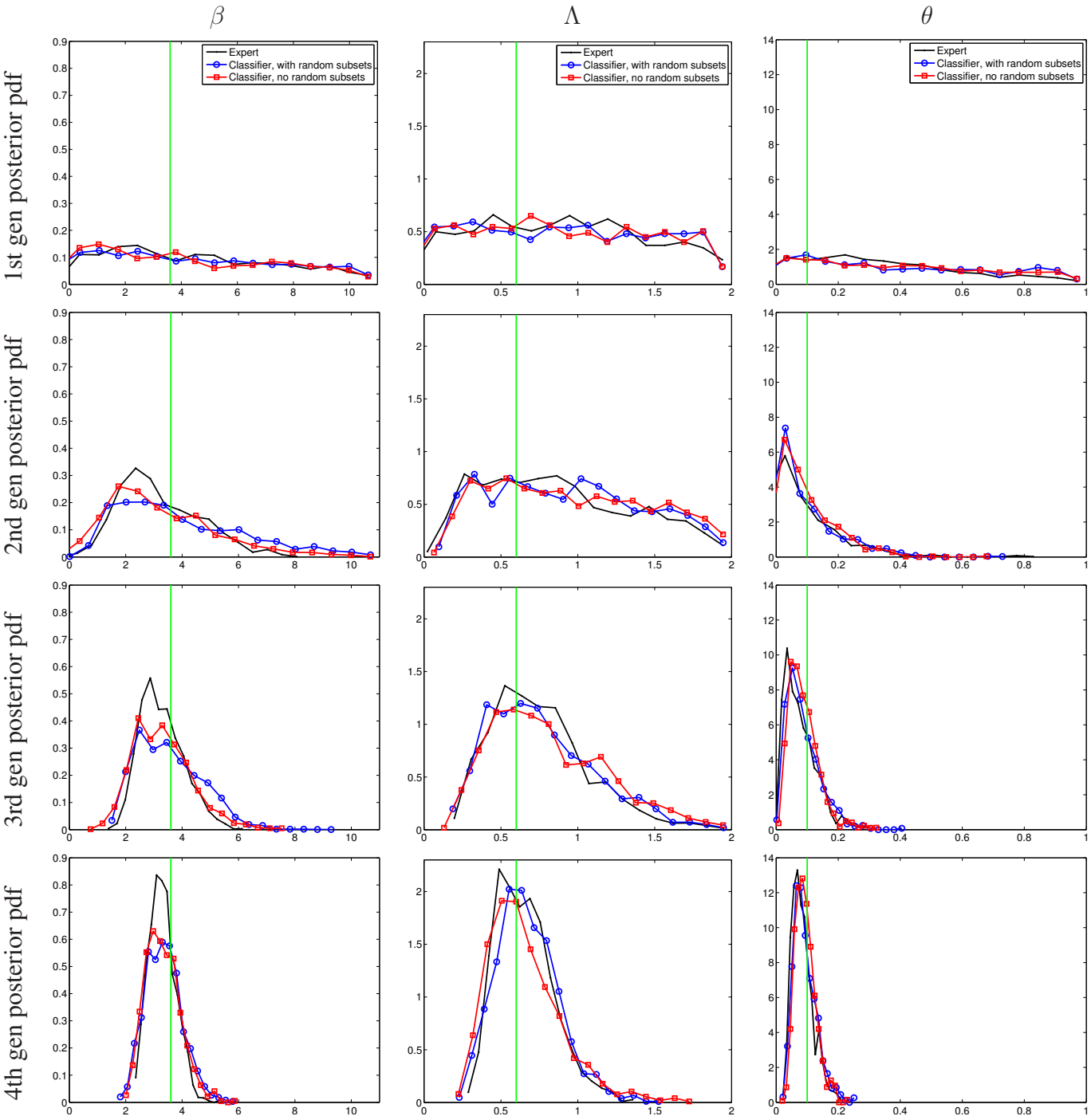
6.3 Further results on compensating missing expert statistics with classifier ABC

Classifier ABC, or more generally the discrepancy measure J_n, is able to incorporate expert statistics, by letting them be features (covariates) in the classification. On the one hand, this allows for expert knowledge to be used in classifier ABC. On the other hand, it allows to enhance expert statistics by data-driven choices. The latter is particularly important if only a insufficient set of summary statistics may be specified. We show here that classifier ABC can counteract shortcomings caused by a suboptimal choice of expert statistics.

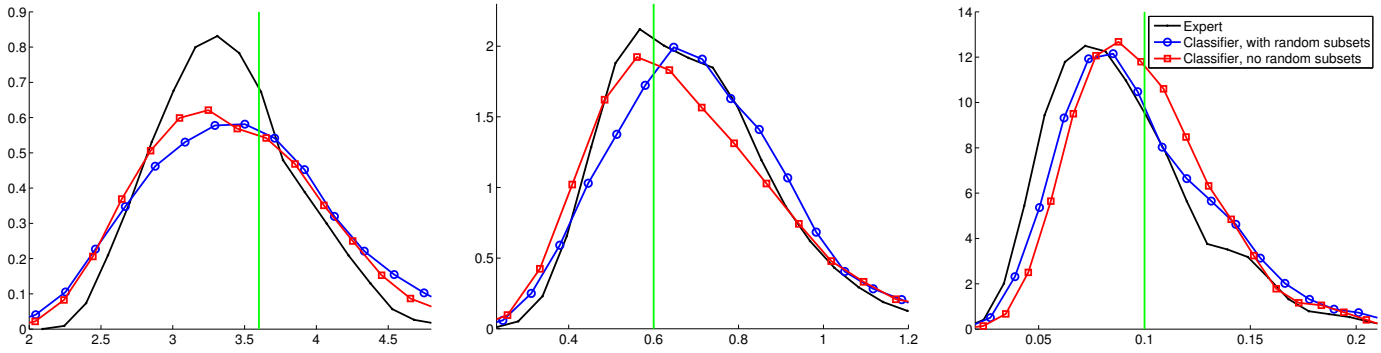
We selected two (simple) expert statistics used by Numminen et al. (2013), namely the number of different strains circulating and the proportion of individuals who are infected. We then inferred the posteriors with this reduced set of summary statistics only, using the method of Numminen et al. (2013). Supplementary Figure 19 visualizes the resulting posterior pdfs (curves in magenta with diamond markers). A comparison with the expert solution with a full set of summary statistics (black curve, point markers) shows that the posterior distributions of Λ and θ are affected by the suboptimal choice of expert statistics. We then included the two selected expert statistics as additional features in classifier ABC. Consequently, the posteriors of Λ and θ recuperated, both when random features were present (cyan curve with triangles) or not (red curve with hexagrams).



Supplementary Figure 12: Testing the applicability of the discrepancy measure J_n to infer the individual-based epidemic model. The figures show $J_n(\boldsymbol{\theta})$ when one parameter is fixed at a time. The red cross marks the parameter value $\boldsymbol{\theta}^o = (\beta^o, \Lambda^o, \theta^o) = (3.6, 0.6, 0.1)$ which we used to generate $\mathbf{X}$. The presence of random features produced more localized regions with small J_n.

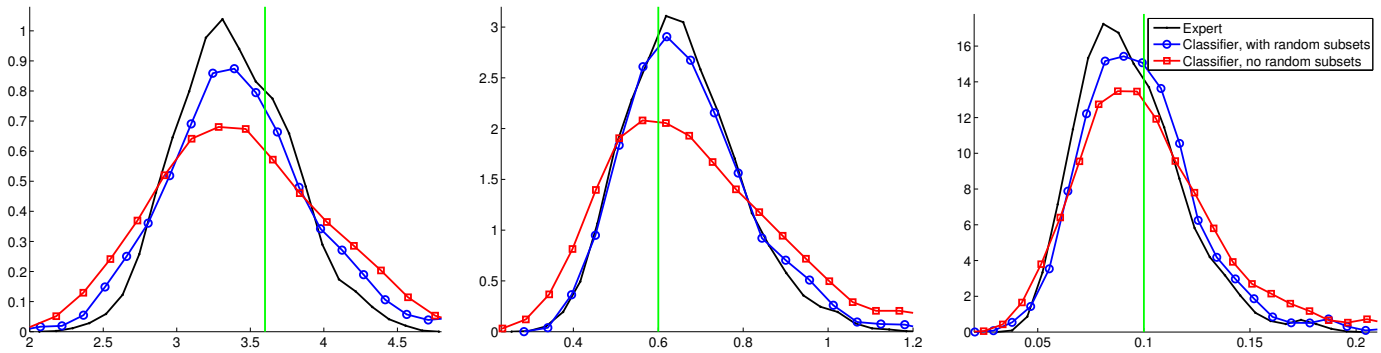


Supplementary Figure 13: Simulated data: Evolution of the posterior pdfs (scaled histograms of the samples). Black, points: ABC solution using expert knowledge, produced with code from Numminen et al. (2013). Blue, circles: classifier ABC with random subsets. Red, squares: classifier ABC without random subsets. Green vertical lines: location of the data generating parameter $\boldsymbol{\theta}^o$. The results are for 1,000 ABC samples.



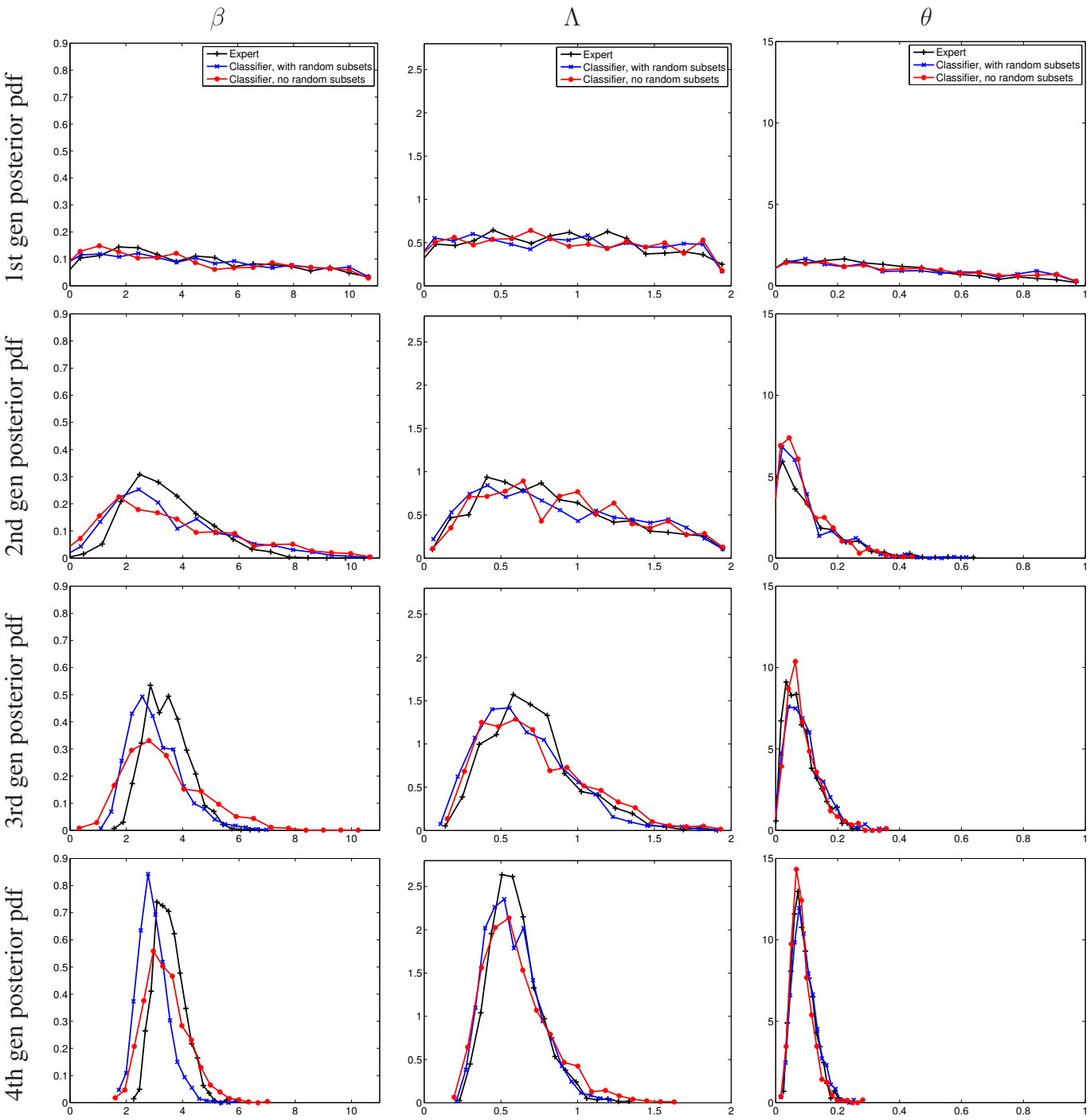
(a) Posterior pdf for β (b) Posterior pdf for Λ (c) Posterior pdf for θ

Supplementary Figure 14: Simulated data: Zoom for the fourth generation in Supplementary Figure 13. The posterior pdf is here a kernel density estimate based on 1,000 ABC samples. We used matlab's ksdensity.m with the default settings, that is, a Gaussian kernel with an adaptively chosen bandwidth. Classifier ABC with random subsets (blue, circles) or without (red, squares) both yielded results which are qualitatively similar to the expert solution (black, points).

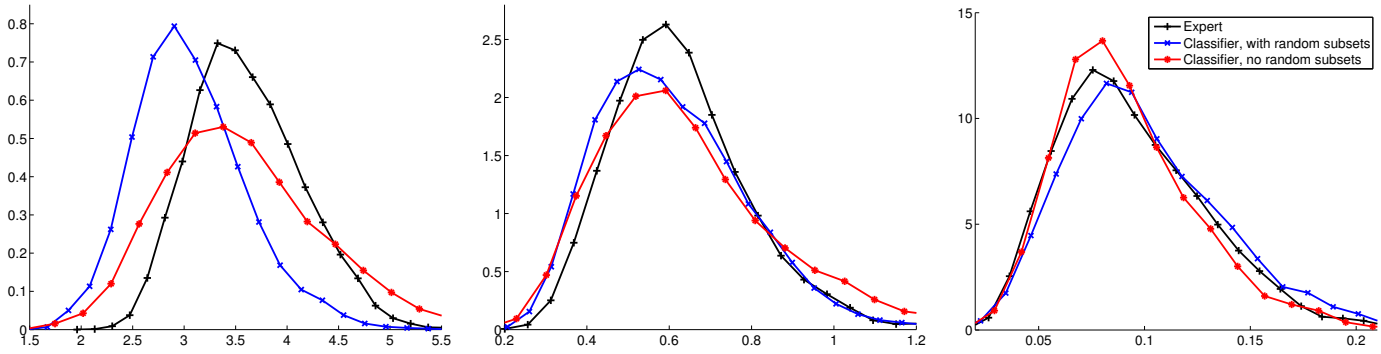


(a) Posterior pdf for β (b) Posterior pdf for Λ (c) Posterior pdf for θ

Supplementary Figure 15: Simulated data: Fifth generation results. Settings and visualization are as in Supplementary Figure 14. In the fifth generation, classifier ABC with random projections (blue, circles) yielded results which are more similar to the expert solution (black, points) than classifier ABC without random projections (red, squares).

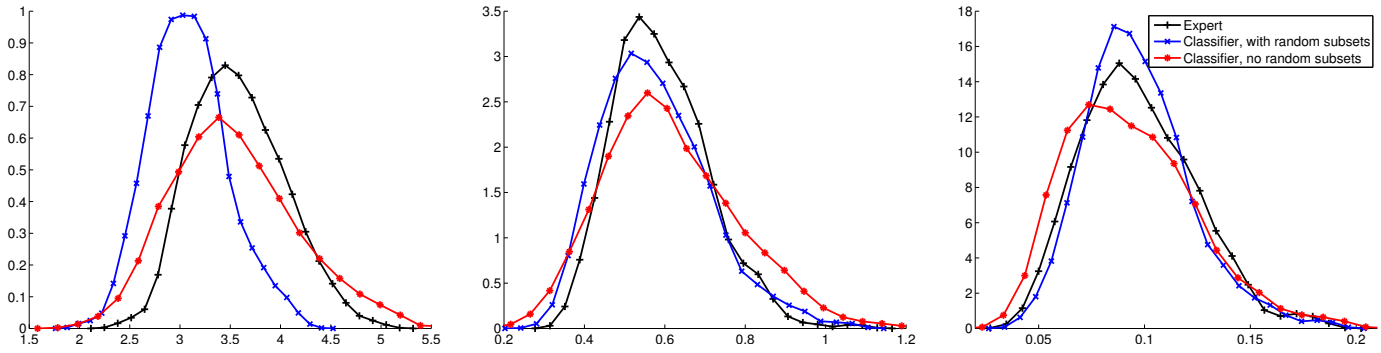


Supplementary Figure 16: Real data: Evolution of the posterior pdfs (scaled histograms of the samples). Black, crosses: ABC solution using expert knowledge, produced with code from Numminen et al. (2013). Blue, crosses: classifier ABC with random subsets. Red, asterisks: classifier ABC without random subsets. The results are for 1,000 ABC samples.



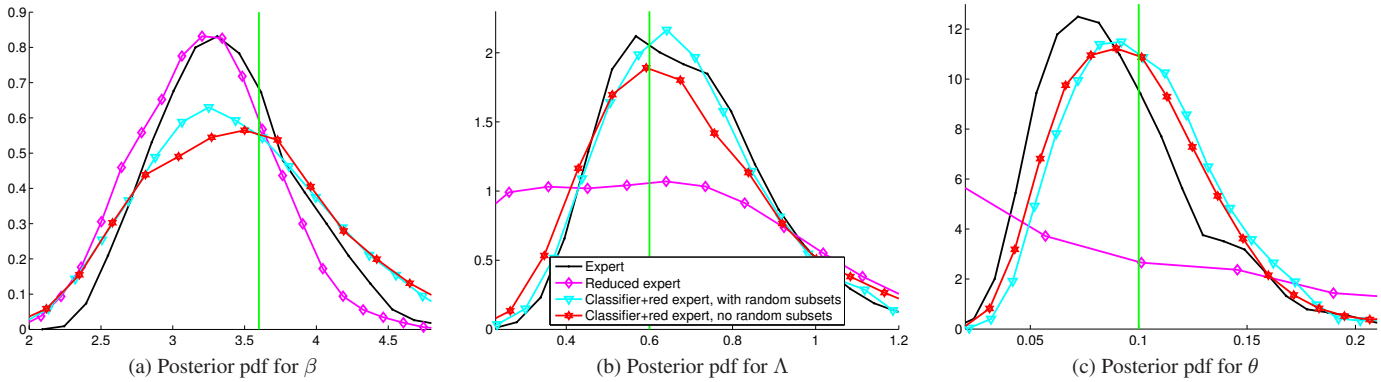
(a) Posterior pdf for β (b) Posterior pdf for Λ (c) Posterior pdf for θ

Supplementary Figure 17: Real data: Zoom for the fourth generation in Supplementary Figure 16. The posterior pdf is here a kernel density estimate based on 1,000 ABC samples. We used matlab's ksdensity.m with the default settings, that is, a Gaussian kernel with an adaptively chosen bandwidth. For Λ and θ, the posteriors for all three methods are qualitatively similar. The posterior of β for classifier ABC with random subsets (blue, crosses) has a smaller mode than the posteriors for classifier ABC without random subsets (red, asterisks) or the expert solution (black, plus markers).



(a) Posterior pdf for β (b) Posterior pdf for Λ (c) Posterior pdf for θ

Supplementary Figure 18: Real data: Fifth generation results. Settings and visualization are as in Supplementary Figure 17. Compared to the fourth generation results in Supplementary Figure 17, the posteriors for classifier ABC with random subsets (blue, crosses) and the expert solution (black, plus markers) are more concentrated than the posterior for classifier ABC without random subsets (red, asterisks).



(a) Posterior pdf for β (b) Posterior pdf for Λ (c) Posterior pdf for θ

Supplementary Figure 19: Using expert statistics in classifier ABC. The results are for simulated data and show the fourth generation pdfs, as in Supplementary Figure 14. ABC with a reduced set of expert statistics affected the posteriors (black curve with points vs magenta curve with diamonds as markers). Classifier ABC was able to counteract the shortcomings caused by the suboptimal choice of expert statistics (cyan curve with triangles and red curve with hexagrams).

References

Beaumont, M., J.-M. Cornuet, J.-M. Marin, and C. Robert (2009). Adaptive approximate Bayesian computation. *Biometrika 96*(4), 983–990.

Fan, R.-E., K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin (2008). Liblinear: A library for large linear classification. *Journal of Machine Learning Research 9*, 1871–1874.

Gelman, A., J. Carlin, H. Stern, and D. Rubin (2003). *Bayesian Data Analysis*. Chapman & Hall.

Gutmann, M. and A. Hyvärinen (2013). Estimation of unnormalized statistical models without numerical integration. In *Proceedings of the Sixth Workshop on Information Theoretic Methods in Science and Engineering (WITMSE)*.

Hastie, T., R. Tibshirani, and J. Friedman (2009). *The Elements of Statistical Learning*. Springer.

Marin, J.-M., P. Pudlo, C. Robert, and R. Ryder (2012). Approximate Bayesian computational methods. *Statistics and Computing 22*(6), 1167–1180.

Numminen, E., L. Cheng, M. Gyllenberg, and J. Corander (2013). Estimating the transmission dynamics of Streptococcus pneumoniae from strain prevalence data. *Biometrics 69*(3), 748–757.

Sisson, S., Y. Fan, and M. Tanaka (2007). Sequential Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences 104*(6), 1760–1765.

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

van der Vaart, A. (1998). *Asymptotic Statistics*. Cambridge University Press.

Wasserman, L. (2004). *All of statistics*. Springer.