Abstract

Statistical modeling of data is an integral part of science. Increasingly complex models are being used as they allow for realistic data generation, but a common difficulty with them is the prohibitively large computational cost to perform likelihood-based inference. In stochastic dynamical systems, for example, latent variables render the likelihood function intractable but simulating observations from the model is feasible. A likelihood-free inference framework has been proposed where the model parameters are identified by finding values yielding simulated data that resemble the observed data. For Bayesian inference, the framework is known as approximate Bayesian computation; for point-estimation generally as indirect inference. We address here one of the open key problems in this framework: the assessment of how similar the observed and simulated data are. Currently, the comparison is often based on subjectively chosen summary statistics and the goal of this paper is to enable a more objective analysis of similarity. Our approach uses the discriminability (classifiability) of the observed and simulated data as discrepancy measure. The problem of choosing an appropriate similarity measure is thereby reduced to a classification problem where we can leverage on effective existing solutions. The validity of our approach is analyzed theoretically and demonstrated for both point estimation and Bayesian inference on synthetic data and real data with an individual-based epidemic model for bacterial infections in day care centers.

Keywords: statistical inference, intractable likelihoods, classification, approximate Bayesian computation
1. Introduction

Statistical inference plays an important role in modern science. A key ingredient of statistical inference is the likelihood function, but evaluating it for complex statistical models can be computationally very costly, which often prevents its use in practice. In this paper, we propose a new computationally feasible principle for statistical inference which does not rely on the likelihood function.\(^1\) It is particularly suited for models with many unobserved variables.

The likelihood function is the probability of the observed data \(X\) under the statistical model. We do not make any particular independence assumptions for the data. The computation of this probability becomes easily intractable if the model is unnormalized or if there are unobserved (latent) variables. For unnormalized models, there exist a number of alternatives to likelihood-based inference (9, 11, 13–15, 17, 18). Making the basic assumption that sampling from the statistical model is possible in a reasonable time, simulation-based inference methods have been proposed which can deal with latent variables (1–3, 5, 6, 8, 12, 19–22, 25, 26, 28–32, 35, 36). In a point-estimation framework, these are generally known as method of simulated moments, or indirect inference, and in a Bayesian framework, as approximate Bayesian computation. We make the same assumption: The principle which is proposed in this paper belongs to the class of simulation-based likelihood-free inference methods.

The basic idea of the simulation-based methods is to infer the vector of parameters \(\theta\) by identifying values for which simulated data \(Y_\theta\), generated with parameter \(\theta\), are close to the observed data \(X\). The similarity between \(Y_\theta\) and \(X\) is usually measured as the distance between some summary statistics of them. Both the distance function used and the summary statistics are critical for the success of the inference procedure. Presently, researchers often choose the two quantities subjectively, relying on expert knowledge. The goal of this paper is to make the analysis of similarity more objective. Benefits of a more objective analysis are that the inference methods become easier to use for non-specialists and that they are more resistant to unfavorable choices of summary statistics. Related previous work includes research on the selection and construction of summary statistics in approximate Bayesian computation (1, 8, 19, 25, 35).

The new principle for likelihood-free statistical inference is motivated by the basic observation that distinguishing two data sets which were generated with very different values of \(\theta\), as in the left panel of Figure 1, is usually easier than discriminating data which were generated with similar values, as in the right panel of Figure 1. Based on this motivation, we here propose to use discrimination (classification) accuracy to measure the discrepancy between the observed and simulated data. The proposed approach reduces the problem of choosing an appropriate distance function and summary statistics to a classification problem, where we can leverage on a wealth of existing solutions. The assessment of similarity becomes thus essentially a data-driven operation. Our results show that the principle may be used for inference in both a point-estimation or Bayesian framework, and that it is applicable to a wide range of different kinds of data, be it continuous, binary, discrete, or a time series.

---

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.
Figure 1: Classification accuracy as discrepancy measure. The model is a bivariate Gaussian with identity covariance matrix. The parameter of interest is the mean $\theta$. Observed data $X$ (black circles) were generated with $\theta^o = (0, 0)$. Left: Simulated data $Y_\theta$ (green diamonds) were generated with $\theta = (6, 0)$. They can be easily distinguished from $X$. The classification rule indicated by the color of the hatched areas yields a classification accuracy of 100%. Right: $Y_\theta$ was generated with $\theta = (1/2, 0)$. Distinguishing such data from $X$ is more difficult; the indicated classification rule has a performance of about 58% correct assignments. When data are generated with $\theta = \theta^o$, the discrimination task cannot be solved significantly above chance-level. We propose likelihood-free inference by identifying parameters which yield chance-level classification performance only.

2. Results

2.1 Classification accuracy as discrepancy measure

The classification accuracy in Figure 1 decreases from 100% (perfect classification performance) towards 50% (chance-level performance) as $\theta$ approaches $\theta^o$, the parameter value which was used to generate the observed data $X$. This suggests inference by identifying parameters which yield chance-level classification performance only.

Low classification accuracies can, however, trivially be obtained by random guessing. It is hence necessary to employ good classification rules. The rule which yields the largest classification accuracy on average consists in assigning a data point to $X$, say, if it is more probable that the datum belongs to $X$ than to $Y_\theta$. The rule is called the Bayes classification rule (BCR). It involves the unknown probability distribution of the data and is thus not known either. But the classification literature provides a wealth of methods to learn an approximation $\hat{h}_\theta$ of the Bayes classification rule $h^*_\theta$ from $X$ and $Y_\theta$. We use several such methods in this paper: linear discriminant analysis (LDA), quadratic discriminant analysis (QDA), L1-regularized polynomial logistic regression, L1-regularized polynomial support vector machine (SVM) classification, and a max-combination of the above and other methods (Max-Rule, see Materials and Methods). These are by no means the only applicable methods. In fact, any method yielding a good approximation of $h^*_\theta$ may be chosen; our paper makes the complete arsenal of classification methods available to parametric inference.

We next show for a range of different kinds of data that most of these methods perform equally well 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 (see Materials and Methods). Further examples are given in Appendix A.1. For the first three data sets, the unknown parameter is the mean. For the moving average model, the lag coefficient is the unknown quantity. 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 single data point. Figure 2 shows that for the Gaussian, Bernoulli, and Poisson data, all considered classification methods perform as well as the Bayes classification rule. Importantly, the parameter $\theta^o$ which we used to generate $X$ can be identified by minimizing the classification accuracy. 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 $\hat{h}^*_{\theta}$ is outside the family of possible classification rules learned by LDA.

For finite data sizes, there is the issue of overfitting in classification. In our case where a large classification accuracy corresponds to a large discrepancy, overfitting means that the two data sets are judged to be less similar than they actually are. To mitigate the effects of overfitting, we use $K$-fold cross-validation: $K$ folds of training and validation sets are extracted from the observed data $X$ and the data simulated with $\theta, Y_\theta$. The training sets are used for learning the classification rules $\hat{h}^k_{\theta}$ whose performances $CA^k$ are then measured on the validation sets. The different validation sets are disjoint. We denote the data points from their union by $y_i$ and $x_i$, $i = 1, \ldots, n$. In the literature, these are typically called features or covariates (16, 23). In many cases, the covariates are equal to the elements of $X$ and $Y_\theta$, but they can also be some transformations of them; for instance, in the moving average example, a pair $(x_t, x_{t+1})$ forms a point $x_i$. The cross-validated classification accuracy $J_n(\theta)$,

$$J_n(\theta) = \frac{1}{K} \sum_{k=1}^{K} CA^k,$$

is used as the measure of discrepancy between $X$ and $Y_\theta$. Being an average, $J_n$ may be interpreted as an estimate of the expected classification accuracy.

2.2 Point estimation

In point estimation, we are interested in finding the single “best” parameter value. In our context, this is the parameter value for which the simulated data $Y_\theta$ are the least distinguishable from the observed data $X$, that is, the parameter $\hat{\theta}_n$ which minimizes $J_n$,

$$\hat{\theta}_n = \text{argmin}_{\theta} J_n(\theta).$$

We show next that the estimator $\hat{\theta}_n$ is consistent. Assuming that the observed data $X$ equal some $Y_{\theta^o}$, generated with unknown parameter $\theta^o$, conditions are given under which $\hat{\theta}_n$ recovers $\theta^o$ as the sample size $n$ increases. Motivating evidence for consistency is provided in Appendix A.2 for the examples in Figure 2.

Some basic assumptions are needed: The data points $x_i$ from $X$ are assumed to have the marginal probability measure $P_{\theta^o}$ and the points $y_i$ from $Y_\theta$ the marginal probability
Figure 2: Comparison of the learned and the Bayes classification rule for large sample sizes 
\( (n = 100,000) \). The symmetric curves depict the classification accuracy for different types of 
data and different classification methods 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.

measure \( P_\theta \) for all \( i \), which amounts to a weak stationarity assumption. Both \( P_{\theta^o} \) and \( P_\theta \) are unknown. We also assume that the parametrization of \( P_\theta \) is not degenerate, that is, there is a compact set \( \Theta \) containing \( \theta^o \) where \( \theta \neq \theta^o \) implies that \( P_\theta \neq P_{\theta^o} \).

The following proposition, proved in Appendix B, lists two conditions for consistency. The first condition is related to convergence of relative frequencies to probabilities (law of large numbers), the second to the ability to learn the Bayes classification rule more accurately as the sample size increases. Convergence in probability as \( n \) increases is denoted by a \( P \) over an arrow.

**PROPOSITION 1** Denote the set of points which the Bayes classification rule \( h_{\theta^o} \) classifies as being from the observed data by \( H_{\theta^o} \), and let \( \mathcal{H}^* = \{ H_\theta, \theta \in \Theta \} \) be a collection of such sets. Denote by \( P_\theta \) the empirical probability measure based on \( n \) points \( y_i \sim P_\theta \), and by \( J_n^*(\theta) \) the classification accuracy for the Bayes classification rule.

\[
\text{If } \sup_{\theta \in \Theta} \sup_{H \in \mathcal{H}^*} |P_\theta(H) - P_\theta^0(H)| \xrightarrow{P} 0 \quad (3)
\]

and

\[
\sup_{\theta \in \Theta} |J_n^*(\theta) - J_n(\theta)| \xrightarrow{P} 0 \quad (4)
\]

then

\[
\hat{\theta}_n \xrightarrow{P} \theta^o. \quad (5)
\]

The collection of sets \( \mathcal{H}^* \) is generated by the different Bayes classification rules \( h_{\theta^o}, \theta \in \Theta \). Equation 3 means that this collection of sets must be (weak) \( P_{\theta^o} \)-Glivenko-Cantelli (34) and that the rate of convergence needs to be equally fast for all \( \theta \) in the neighborhood of \( \theta^o \). This is a condition on the “size” of \( \mathcal{H}^* \), and hence the complexity of the Bayes classification rules.
In other words, the condition in Equation 3 is about the difficulty of the classification problems. The condition in Equation 4, 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 is a research area in itself (37). In Figure 2, the curves for the different learning rules are practically the same as those for the optimal Bayes classification rule (with the exception of LDA on the moving average data), which suggests that Equation 4 holds for the examples in that figure, in line with the results shown in Appendix A.2.

2.3 Approximate Bayesian computation

Approximate Bayesian computation (ABC) comprises several simulation-based methods to obtain posterior samples when the likelihood function is not known, see, for example, (20) for a review. A major difficulty in ABC is the measurement of discrepancy between observed and simulated data. 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 Materials and Methods). The use of $J_n$ in ABC is, however, not restricted to this particular algorithm.

Empirical posterior probability density functions (pdfs) inferred by classifier ABC for binary (Bernoulli), count (Poisson), continuous (Gaussian), and time-series (ARCH) data are shown in Figure 3. The figure also contains the true posterior distributions (red-solid curves) for reference. The true posterior for the ARCH data is not available in closed form. We approximated it using deterministic numerical integration, as detailed in the supplementary methods in Appendix C.1. There is a good match between the inferred and the reference posterior distributions. Different classification methods yield different results but the overall performance is rather similar, see the supplementary results in Appendix A.3 for further evidence. 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 which show the evolution of the posterior samples in the ABC algorithm are provided in Appendix A.3.

As a quantitative analysis, we computed the relative error of the posterior means and standard deviations. The results, reported in Appendix A.3, 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. The relative errors of the posterior standard deviations are all positive which means that the inferred posteriors have a larger dispersion than the reference posteriors.

2.4 Application in epidemiology

We next applied classifier ABC to a stochastic individual-based epidemic model for bacterial infections in day care centers. The observed data $X$ are a random sample of 29 binary matrices of different sizes. Each matrix represents a day care center, 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 model was developed in (24) and has three parameters: Parameter $\beta$ which is related to the probability to be infected by someone inside a day care center, parameter $\Lambda$ for the probability of an infection from an outside source, and parameter $\theta$ which is related to the probability to be infected with multiple strains. In (24), inference was performed with ABC. The summary statistics were chosen based on epidemiological considerations and the distance function was adapted to the specific problem at hand. Materials and Methods and Appendix C.3 contain a brief review of the model and the inference method of (24).

Standard classifiers operate on features in vector form, and hence for inference with classifier ABC, the observed matrices needed to be transformed to feature vectors. We used
simple standard features which reflect the matrix structure and the binary nature of the data: rank and norm of the singular values of the matrices were used as features, as well as average and standard deviation of the fraction of ones in certain subsets of each matrix. We used the row and column subsets, as well as randomly chosen ones (see Materials and Methods).

In ABC, the applicability of the chosen summary statistics can be assessed by first performing inference on synthetic data which were simulated from the model. We can use simulated data in the same way to assess the features and classifiers. Our analysis, reported in Appendix A.4, revealed that LDA, the arguably simplest classification method, is suitable to infer the epidemic model. Furthermore, the required computation time was about the same as for the expert method (on average, the total time for the data generation and the discrepancy measurement was 28.49 ± 3.45 seconds for LDA while it was 28.41 ± 3.45 seconds for the expert method; with 28.4 ± 3.45 seconds, most of the time was spent on generating data from the epidemic model).

Figure 4 shows the inferred posterior distributions. As in (24), the results were obtained with a sequential Monte Carlo ABC algorithm with four generations. Classifier ABC yields similar results as the expert solution, on both simulated and real data. For the real data, the posterior mode of $\beta$ is slightly smaller for classifier ABC. This point is further discussed in Appendix A.4. There, we also show that omitting the random subsets yields posterior distributions with a larger variance.

A further advantage of the proposed data-driven approach is its compatibility with expert statistics. Classifier ABC, and more generally the discrepancy measure $J_n$, is able to incorporate expert statistics by letting them be features (covariates) in the classification. Combining expert statistics and classifier ABC can be worthwhile if the available expert statistics alone are insufficient to perform ABC: We report in Appendix A.4 that classifier ABC was able to compensate for missing expert statistics.

![Figure 4](image.png)

Figure 4: Classifier ABC on a stochastic individual-based epidemic model. The results are for real (solid curves) and simulated data (dashed curves). The posterior pdfs shown are kernel density estimates (smoothed and scaled histograms) based on 1,000 ABC samples. The expert solution with the method from (24) 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 line indicates the parameters used to generated the simulated observed data.
3. Discussion

Likelihood-free inference is a powerful methodology which can be applied to realistic statistical models. A limiting difficulty is the required discrepancy measurement between simulated and observed data. We found that classification accuracy can be used as a discrepancy measure. This finding has practical value because it reduces the difficult problem of choosing an appropriate discrepancy measure to a more standard problem; 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 different at first glance, are actually tightly connected.

Using classification accuracy as a discrepancy measure is a data-driven approach to assess the similarity between simulated and observed data. It is, however, not necessarily a black-box approach. Knowledge about the problem at hand can be incorporated when specifying the classification problem. For the moving average model, for example, knowledge about the statistical dependencies in the data led us to choose two consecutive time points as features (covariates) in the classification. In our application to epidemiology, we showed that expert statistics can be incorporated in the form of features. These may be subjective choices but the advantage of the classifier approach is that the influence of the choices is rather weak because the importance of the features is automatically determined with the learning of the classification rules. This property of the proposed approach allowed us to even use random features in the inference of the epidemic model.

Further exploration of the connection between likelihood-free inference and classification may lead to practical improvements of our current approach. Each parameter \( \theta \) induces a classification problem. We treated them separately in this paper. However, they are actually related: First, the observed data \( X \) occur in all the classification problems. Second, the simulated data sets \( Y_\theta \) 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 on the solution of one problem to solve another one is generally known as transfer learning (27). In the same spirit, leveraging on transfer learning, or other methods from classification, seems promising to further advance likelihood-free inference.

4. Materials and Methods

4.1 Models

We tested our method on data of different types. Details are provided in Appendix C.1. In brief, for the examples with independent data points, the Gaussian distribution was used as a representative of continuous-valued data. We inferred either only the mean or both the mean and variance. For binary data, we used Bernoulli, and for count data Poisson random variables. In all these cases conjugate priors were used for Bayesian inference. For time series data, we considered two examples: The first one was a zero mean moving average model with an unknown lag coefficient. For Bayesian inference, a uniform prior was used. The \( x_i \) for classification consisted of 2 consecutive time points \((x_t, x_{t+1})\). The second example was an ARCH(1) model, see for example (4), with unknown mean and variance process coefficients, and uniform priors. The \( x_i \) used for classification consisted
of 5 consecutive time points. The posterior distribution cannot be computed analytically in closed form in either of the two time series examples. We used deterministic numerical integration to approximate it, as detailed in Appendix C.1. For all examples, \( n \) was 50 for the results in the ABC section.

### 4.2 Classifiers

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, for example, (16, 23). We used methods provided by two libraries: For linear and quadratic discriminant analysis (LDA and QDA), matlab’s \texttt{classify.m} was employed. For \( L_1 \) and \( L_2 \) regularized polynomial logistic regression and support vector machine (SVM) classification, we used the \texttt{liblinear} classification library (7), version 1.93, via the matlab interface, with a fixed regularization penalty (we used the default value \( C = 1 \)). The \texttt{liblinear} library is for linear classification. Polynomial classification was implemented via polynomial basis expansion (16, 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 \( x_i \) were projected onto their principal components prior to classification and thereafter rescaled to variance one. This operation amounts to multiplying the \( x_i \) with a whitening matrix, and the \( y_i \) were multiplied with the same matrix. For cross-validation, \( K = 5 \) folds were used. The max-rule consisted in trying several classifiers 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 classifiers used for the max-rule.

### 4.3 ABC algorithm

There are several algorithms for approximate Bayesian computation (ABC), see (20) for an overview. The algorithms are iterative: The basic steps at each iteration are proposing a parameter \( \theta' \), simulating a pseudo observed data set \( Y_{\theta'} \), and then accepting or rejecting the proposal based on a comparison of \( Y_{\theta'} \) with the actually observed data \( X \). The particular way the comparison is performed is left to the user. Our discrepancy measure \( J_n \) implements one way to perform the comparison. Its use is not tied to a particular ABC algorithm. For the results in this paper, we used a population Monte Carlo sampler, also known as sequential Monte Carlo ABC algorithm, with a Gaussian kernel (20, Algorithm 4), (3, 29, 32). 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 chose the threshold as the maximum of the value from a pre-defined schedule and the 0.1 quantile of the $J_n$ from the previous generation. Appendix C.2 contains further details and a brief analysis of this approach.

4.4 Application in epidemiology

The stochastic individual-based epidemic model of (24) is a continuous-time Markov chain for the transmission dynamics inside a day care center, see Appendix C.3 for further details. Uniform priors were assumed for the three parameters of the model: $\beta \in (0,11), \Lambda \in (0,2),$ and $\theta \in (0,1)$. In previous work, the model was inferred with ABC where the discrepancy between $X$ and $Y_{\theta}$ was assessed using expert-knowledge rooted in epidemiology (24). 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 $X$ and $Y_{\theta}$. Appendix C.3 contains a brief review of the inference method of (24).

In classifier ABC, we extracted features $x_i$ and $y_i$ from the observed and simulated data. Since the data were binary and matrix-valued, we chose features related to these two properties. 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. As subsets, the set of rows and the set of columns were used, as well as 100 randomly chosen ones. 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. Each datum $x_i$ or $y_i$ 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 $n = 1,000$ data points $x_i$ and $y_i$. We also ran classifier ABC without random subsets; the classification problems consisted then in discriminating between two data matrices of size $5 \times 29$ only.

Acknowledgments

MUG and RD wish to thank Elina Numminen for providing computer code for the epidemic model. 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).

Author contributions: MUG proposed and designed research; MUG and RD performed research and analyzed data; MUG wrote the paper; RD, SK, and JC assisted in research design and writing.

References

[1] S. Aeschbacher, M.A. Beaumont, and A. Futschik. A novel approach for choosing summary statistics in approximate Bayesian computation. *Genetics*, 192(3):1027–1047, 2012.
[2] M.A. Beaumont, W. Zhang, and D.J. Balding. Approximate Bayesian computation in population genetics. *Genetics*, 162(4):2025–2035, 2002.

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

[4] A.K. Bera and M.L. Higgins. ARCH model: Properties, estimation and testing. *Journal of Economic Surveys*, 7(4):305–366, 1993.

[5] P.J. Diggle and R.J. Gratton. Monte Carlo methods of inference for implicit statistical models. *Journal of the Royal Statistical Society. Series B (Methodological)*, 46(2):193–227, 1984.

[6] C.C. Drovandi, A.N. Pettitt, and M.J. Faddy. Approximate Bayesian computation using indirect inference. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 60(3):317–337, 2011.

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

[8] P. Fearnhead and D. Prangle. Constructing summary statistics for approximate Bayesian computation: semi-automatic approximate Bayesian computation. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 74(3):419–474, 2012.

[9] A. Gelman. Method of moments using Monte Carlo simulation. *Journal of Computational and Graphical Statistics*, 4(1):36–54, 1995.

[10] A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin. *Bayesian Data Analysis*. Chapman & Hall, 2003.

[11] C.J. Geyer. On the convergence of Monte Carlo maximum likelihood calculations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 56(1):261–274, 1994.

[12] C. Gouriéroux, A. Monfort, and E. Renault. Indirect inference. *J. Appl. Econ.*, 8(S1):S85–S118, 1993.

[13] M.U. Gutmann and J. Hirayama. Bregman divergence as general framework to estimate unnormalized statistical models. In *Proc. Conf. on Uncertainty in Artificial Intelligence (UAI)*, pages 283–290, Corvallis, Oregon, 2011. AUAI Press.

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

[15] M.U. Gutmann and A. Hyvärinen. Estimation of unnormalized statistical models without numerical integration. In *Proc Workshop on Information Theoretic Methods in Science and Engineering*, 2013.

[16] T. Hastie, R. Tibshirani, and J.H. Friedman. *The Elements of Statistical Learning*. Springer, 2009.
[17] G.E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.

[18] A. Hyvärinen. Estimation of non-normalized statistical models using score matching. *Journal of Machine Learning Research*, 6:695–709, 2005.

[19] P. Joyce and P. Marjoram. Approximately sufficient statistics and Bayesian computation. *Statistical Applications in Genetics and Molecular Biology*, 7(1):Article 26, 2008.

[20] J.-M. Marin, P. Pudlo, C. Robert, and R. Ryder. Approximate Bayesian computational methods. *Statistics and Computing*, pages 1–14, 2011.

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

[22] D. McFadden. A method of simulated moments for estimation of discrete response models without numerical integration. *Econometrica*, 57(5):995–1026, 1989.

[23] K.P. Murphy. *Machine Learning, A Probabilistic Perspective*. MIT Press, 2012.

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

[25] M.A. Nunes and D.J. Balding. On optimal selection of summary statistics for approximate Bayesian computation. *Statistical Applications in Genetics and Molecular Biology*, 9(1):Article 34, 2010.

[26] A. Pakes and D. Pollard. Simulation and the asymptotics of optimization estimators. *Econometrica*, 57(5):1027–1057, 1989.

[27] S.J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[28] J.K. Pritchard, M.T. Seielstad, A. Perez-Lezaun, and M.W. Feldman. Population growth of human Y chromosomes: a study of Y chromosome microsatellites. *Molecular Biology and Evolution*, 16(12):1791–1798, 1999.

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

[30] A.A. Smith. *The New Palgrave Dictionary of Economics*, chapter Indirect Inference. Palgrave Macmillan (London), 2nd edition, 2008.

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

[32] T. Toni, D. Welch, N. Strelkowa, A. Ipsen, and M.P.H Stumpf. Approximate Bayesian computation scheme for parameter inference and model selection in dynamical systems. *Journal of the Royal Society Interface*, 6(31):187–202, 2009.
[33] A.W. van der Vaart. *Asymptotic Statistics*. Cambridge University Press, 1998.

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

[35] D. Wegmann, C. Leuenberger, and L. Excoffier. Efficient approximate Bayesian computation coupled with Markov chain Monte Carlo without likelihood. *Genetics*, 182(4):1207–1218, 2009.

[36] S.N. Wood. Statistical inference for noisy nonlinear ecological dynamic systems. *Nature*, 466(7310):1102–1104, 2010.

[37] T. Zhang. Statistical behavior and consistency of classification methods based on convex risk minimization. *The Annals of Statistics*, 32(1):56–85, 2004.
Appendix A. Supplementary Results

A.1. Classification accuracy as discrepancy measure

In Figure 2 in the main text, the minimal classification accuracy, namely chance-level classification performance 0.5, was attained at a point close to the parameter $\theta^o$ which was used to generate $X$. We provide here two more such examples: Figure 5 shows the results for a Gaussian distribution with unknown mean and variance, and Figure 6 the results for the autoregressive conditional heteroskedasticity (ARCH) time series model in Equation 41 with unknown mean and variance process coefficients. Parameter $\theta^o$ is marked with a red cross.

![Contour plots for different classifiers](image_url)

Figure 5: Gaussian with unknown mean and variance. The contour plots show the classification accuracy as a function of the two parameters for large samplesizes ($n = 100,000$). The different panels depict results for different classifiers. All obtain their minimal classification accuracy, chance-level performance 0.5, close to $\theta^o$. 

Figure 6: ARCH(1) model in Equation 41 with unknown mean and variance process coefficients $\theta_1$ and $\theta_2$. The results are for $n = 10,000$ and visualized as in Figure 5.
A.2. Point estimation

We computed the mean squared estimation error $E[||\hat{\theta}_n - \theta^o||^2]$ for different sample sizes $n$ for the examples in Figure 2 in the main text. The mean squared error is plotted against the sample size on a log-log scale in Figure 7. The decay is linear which suggests convergence in quadratic mean, hence convergence in probability, and thus consistency of $\hat{\theta}_n$.

Figure 7: The mean squared estimation error 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. 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 decay on the log-log scale suggests convergence in quadratic mean, and hence consistency of the estimator $\hat{\theta}_n$. 
A.3. Approximate Bayesian computation

This section contains further results for classifier ABC on data with known properties. The related materials and methods are provided in the main text and in Appendix C.1.

A.3.1 The inferred posterior distributions for all classifiers used

We report the posterior distributions for all classifiers used in the paper in Figure 8 to Figure 13. The results are organized according to the modality of the data.

The results are for 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 a contour plot (with red dashed lines) or, if uniform, by hatching its domain.

A.3.2 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 an approximate posterior distribution. The following table contains links to movies which show this process. They are also available at http://www.cs.helsinki.fi/u/gutmann/material/CLAD/movies/.

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

A.3.3 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 the supplementary methods in Appendix C.1.

Figure 14 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.
Figure 8: Binary data: Inferred posterior distribution of the success probability of a Bernoulli random variable.

Figure 9: Count data: Inferred posterior distribution of the mean of a Poisson random variable.
Figure 10: Continuous data: Inferred posterior distribution of the mean of a Gaussian random variable with known variance.

Figure 11: Continuous data: Inferred posterior distribution of the mean and variance of a Gaussian random variable.
Figure 12: Time series: Inferred posterior distribution of the lag coefficient of a zero mean moving average model of order one.

Figure 13: Time series: Inferred posterior distribution of the mean and variance process coefficients of a ARCH(1) model.
Figure 14: 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.
A.4. Application in epidemiology

This section contains further results and a more detailed analysis of our application in epidemiology. The related materials and methods are provided in the main text and in Appendix C.3.

A.4.1 Preliminary investigation on simulated data

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

Since ABC algorithms are rather time consuming, we first tested the applicability in the framework of point estimation. For that purpose, we computed $J_n(\theta)$ 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 $\theta$ the same random seed when simulating the $Y_{\theta}$. The random seeds for $X$ and the $Y_{\theta}$ were different.

Figure 15 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 $X$ and $Y_{\theta}$ 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 $Y_{\theta}$ were generated with different random seeds). Figure 16 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 from (24) using expert-knowledge are shown in black (point markers). The results for the fourth generation are shown separately in Figure 17. The different solutions are qualitatively very similar, even though the tails of the posterior of $\beta$ are heavier for classifier ABC than for the method of (24).

In (24), posterior distributions for four generations are shown. In both the results reported in Figure 16 and the results in (24), 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 Figure 18. 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 Figure 15.

A.4.2 Further results on real data

The evolution of the posterior pdfs during the ABC algorithm is shown in Figure 19. Starting from a uniform distribution, a posterior distribution with a well defined mode emerged. The evolution of the posterior pdfs during the ABC algorithm is shown in Figure 19. Starting from a uniform distribution, a posterior distribution with a well defined mode emerged. The results for the fourth generation are shown in more detail in Figure 20. While the posteriors of $\Lambda$ and $\theta$ are qualitatively similar for all three methods, the posterior of $\beta$ 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 Figure 21. 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 $\beta$ 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.

A.4.3 Using expert statistics in 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 (24), 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 code by (24). Figure 22 visualizes the resulting posterior pdfs (curve 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 $\Lambda$ and $\theta$ 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 $\Lambda$ and $\theta$ recuperated, both when random features were present (cyan curve with triangles) or not (red curve with hexagrams).
Figure 15: A preliminary investigation on simulated $X$ to test the applicability of LDA and the chosen features. The figures show $J_n(\theta)$ when one parameter is fixed at a time. Data $Y_\theta$ are judged to be very different from $X$ if $J_n(\theta) = 1$ while $J_n(\theta) = 1/2$ means that $Y_\theta$ and $X$ are judged to be similar. The red cross marks the location of the parameter $\theta^o = (3.6, 0.6, 0.1)$ which we used to generate $X$. The presence of random features produced slightly more localized regions with small $J_n$. 

$\beta = 3.6$, $\Lambda = 0.6$, $\theta = 0.1$
Figure 16: Simulated data: Evolution of the posterior pdfs (scaled histograms of the samples). Black, points: ABC solution using expert knowledge, produced with code from (24). Blue, circles: classifier ABC with random subsets. Red, squares: classifier ABC without random subsets. Green vertical lines: location of the data generating parameter $\theta^o$. The results are for 1,000 ABC samples.
Figure 17: Simulated data: Zoom for the fourth generation in 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. Classifier ABC with random subsets (blue, circles) or without (red, squares) both yielded results which are qualitatively similar to the expert solution (black, points).

Figure 18: Simulated data: Fifth generation results. Settings and visualization are as in Figure 17. 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).
Figure 19: Real data: Evolution of the posterior pdfs (scaled histograms of the samples). Black, crosses: ABC solution using expert knowledge, produced with code from (24). Blue, crosses: classifier ABC with random subsets. Red, asterisks: classifier ABC without random subsets. The results are for 1,000 ABC samples.
Figure 20: Real data: Zoom for the fourth generation in Figure 19. 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 $\Lambda$ and $\theta$, the posteriors for all three methods are qualitatively similar. The posterior of $\beta$ 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).

Figure 21: Real data: Fifth generation results. Settings and visualization are as in Figure 20. Compared to the fourth generation results in Figure 20, 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).
Figure 22: Using expert statistics in classifier ABC. The results are for simulated data and show the fourth generation pdfs, as in Figure 17. 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).
Appendix B. 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 \( \theta^o \) (33). The proof has three steps: First, we identify \( J \). Second, we find conditions under which \( J \) is minimized by \( \theta^o \). Third, we derive conditions which imply that \( J_n \) converges to \( J \).

B.1. Definition of \( J \)

In Equation 1 in the main text, \( J_n \) is defined as the cross-validated classification accuracy,

\[
J_n(\theta) = \frac{1}{K} \sum_{k=1}^{K} CA^k. \tag{6}
\]

\( CA^k \) is the classification accuracy for the learned classification rule \( \hat{h}^k \) on validation set \( k \), which we here denote by \( D_{val}^k \). Each validation set \( D_{val}^k \) consists of \( m \) data points \( x_i^k \) and \( y_i^k \), \( i = 1, \ldots, m \), extracted from \( X \) and \( Y_\theta \). The different validation sets are disjoint. Associating the data points from \( X \) with class zero, and the points from \( Y_\theta \) with class one, \( CA^k \) can be expressed in terms of \( \hat{h}^k \) and the \( x_i^k \), \( y_i^k \),

\[
CA^k = \frac{1}{2m} \left( \sum_{i=1}^{m} \hat{h}^k(y_i^k) + [1 - \hat{h}^k(x_i^k)] \right) = \frac{1}{2} + \frac{1}{2m} \sum_{i=1}^{m} \hat{h}^k(y_i^k) - \hat{h}^k(x_i^k). \tag{7}
\]

We thus obtain

\[
J_n(\theta) = \frac{1}{K} \sum_{k=1}^{K} \left( \frac{1}{2} + \frac{1}{2m} \sum_{i=1}^{m} \hat{h}^k(y_i^k) - \hat{h}^k(x_i^k) \right) = \frac{1}{2} + \frac{1}{2Km} \sum_{i=1}^{m} \sum_{k=1}^{K} \hat{h}^k(y_i^k) - \hat{h}^k(x_i^k). \tag{8}
\]

Each data point is used exactly once for validation since the \( D_{val}^k \) are disjoint. We make the simplifying assumption that splitting the original \( n \) data points into \( K \) folds of \( m \) points was possible without remainders. We can then order the \( y_i^k \) as

\[
y_1^1, \ldots, y_m^1, y_1^2, \ldots, y_m^2, y_1^3, \ldots, y_m^K,
\]

and relabel them from 1 to \( n \). Doing the same for the \( x_i^k \) we obtain

\[
J_n(\theta) = \frac{1}{2} + \frac{1}{2n} \sum_{i=1}^{n} \hat{h}^{k(i)}(y_i) - \frac{1}{2n} \sum_{i=1}^{n} \hat{h}^{k(i)}(x_i). \tag{9}
\]

The function \( k(i) \) in the equation indicates to which validation set data point \( i \) belonged. The data are used in two ways: As part of the sample average and for the learning of the classification rules.

The Bayes classification rule \( h^*_\theta \) maps a point to zero if the point is more probable under \( P_{\theta^o} \) than under \( P_{\theta} \), and to one otherwise. Since both measures are unknown, the Bayes classification rule is not available. If it were and we used it instead of the \( \hat{h}^k \), we obtained \( J^*_n \),

\[
J^*_n(\theta) = \frac{1}{2} + \frac{1}{2n} \sum_{i=1}^{n} h^*_\theta(y_i) - \frac{1}{2n} \sum_{i=1}^{n} h^*_\theta(x_i). \tag{10}
\]
The data are here only used in the sample average unlike for $J_n$. The normalized sums in the equation are the fractions of points $x_i$ and $y_i$ which $h^*_\theta$ maps to one. The function $k(i)$ disappeared because we made the weak stationarity assumption in the main text that the marginal distributions of $x_i$ and $y_i$ do not depend on $i$.

In what follows, it is helpful to introduce the set $H^*_\theta = \{u: h^*_\theta(u) = 1\}$. The fractions in Equation 10 are then the fractions of points in the validation set which belong to $H^*_\theta$.

Taking the expectation over $X$ and $Y_\theta$, using that the expectation over the binary function $h^*_\theta$ equals the probability of the set $H^*_\theta$,

$$E(h^*_\theta(y_i)) = P_\theta(H^*_\theta), \quad E(h^*_\theta(x_i)) = P_{\theta^*}(H^*_\theta),$$

we obtain $E(J_n^*(\theta)) = J(\theta)$,

$$J(\theta) = \frac{1}{2} + \frac{1}{2} (P_\theta(H^*_\theta) - P_{\theta^*}(H^*_\theta)).$$

The difference between $J_n$ and $J$ is twofold: First, relative frequencies instead of probabilities occur. Second, learned classification rules instead of the Bayes classification rule are used. In step 3 of our proof, we derive conditions under which the relative frequencies converge to probabilities (law of large numbers), and the learned classification rule to the Bayes classification rule. Prior to that, we first show that $J$ is a meaningful limit, in the sense that its minimization allows to identify $\theta^*$. 

**Remark.** There is an interesting analogy between the objective $J_n^*$ and the log-likelihood: The sum over the $y_i$ does not depend on the observed data but on $\theta$ and may be considered an analogue to the log-partition function (or an estimate of it). In the same analogy, the sum over the $x_i$ 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 in (15).

**B.2. Minimization of $J$**

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

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

Restricting the parameter space to $\Theta$, consistency of $\hat{\theta}_n$ follows from uniform convergence of $J_n$ to $J$ on $\Theta$ (33, Theorem 5.7).

**B.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.


PROPOSITION 3

If

\[ \sup_{\theta \in \Theta} |J(\theta) - J_n^*(\theta)| \xrightarrow{P} 0 \tag{13} \]

\[ \sup_{\theta \in \Theta} |J_n^*(\theta) - J_n(\theta)| \xrightarrow{P} 0 \tag{14} \]

then

\[ \sup_{\theta \in \Theta} |J(\theta) - J_n(\theta)| \xrightarrow{P} 0 \tag{15} \]

Proof. By the triangle inequality, we have

\[ |J(\theta) - J_n(\theta)| = |J(\theta) - J_n^*(\theta) + J_n^*(\theta) - J_n(\theta)| \leq |J(\theta) - J_n^*(\theta)| + |J_n^*(\theta) - J_n(\theta)|. \tag{16} \]

Hence

\[ \sup_{\theta \in \Theta} |J(\theta) - J_n(\theta)| \leq \sup_{\theta \in \Theta} |J(\theta) - J_n^*(\theta)| + \sup_{\theta \in \Theta} |J_n^*(\theta) - J_n(\theta)|, \tag{18} \]

which means that the event

\[ \sup_{\theta \in \Theta} |J(\theta) - J_n(\theta)| > \epsilon \]

is included in the event

\[ \left( \sup_{\theta \in \Theta} |J(\theta) - J_n^*(\theta)| + \sup_{\theta \in \Theta} |J_n^*(\theta) - J_n(\theta)| \right) > \epsilon, \]

which implies that

\[ P \left( \sup_{\theta \in \Theta} |J(\theta) - J_n(\theta)| > \epsilon \right) \leq P \left( \sup_{\theta \in \Theta} |J(\theta) - J_n^*(\theta)| + \sup_{\theta \in \Theta} |J_n^*(\theta) - J_n(\theta)| > \epsilon \right). \tag{19} \]

It further holds that

\[ P \left( \sup_{\theta \in \Theta} |J(\theta) - J_n^*(\theta)| + \sup_{\theta \in \Theta} |J_n^*(\theta) - J_n(\theta)| > \epsilon \right) \leq P \left( \sup_{\theta \in \Theta} |J(\theta) - J_n^*(\theta)| > \frac{\epsilon}{2} \right) + P \left( \sup_{\theta \in \Theta} |J_n^*(\theta) - J_n(\theta)| > \frac{\epsilon}{2} \right). \tag{20} \]

which concludes the proof. \( \square \)

The second condition in Proposition 3 above is the condition in Equation 4 in Proposition 1. We next show that Equation 3 in Proposition 1 guarantees that the first condition in Proposition 3 holds.

PROPOSITION 4

If

\[ \sup_{\theta \in \Theta} \sup_{H \in \mathcal{H}^*} \left| P_\theta(H) - P_n^\theta(H) \right| \xrightarrow{P} 0 \tag{21} \]

then

\[ \sup_{\theta \in \Theta} |J(\theta) - J_n^*(\theta)| \xrightarrow{P} 0 \tag{22} \]
**Proof.** By definition of $J$ and $J^*_n$, we have

$$|J(\theta) - J^*_n(\theta)| \leq \frac{1}{2} |P_{\theta}(H^*_\theta) - P^n_{\theta}(H^*_\theta)| + \frac{1}{2} |P_{\theta^0}(H^*_\theta) - P^n_{\theta^0}(H^*_\theta)|. \quad (23)$$

Hence

$$\sup_{\theta \in \Theta} |J(\theta) - J_n(\theta)| \leq \frac{1}{2} \sup_{\theta \in \Theta} |P_{\theta}(H^*_\theta) - P^n_{\theta}(H^*_\theta)| + \frac{1}{2} \sup_{\theta \in \Theta} |P_{\theta^0}(H^*_\theta) - P^n_{\theta^0}(H^*_\theta)|, \quad (24)$$

where the second term depends on $\theta$ via the set $H^*_\theta$. The supremum over $\theta$ can there be replaced by a supremum over the elements of the collection of sets $\mathcal{H}^*$,

$$\sup_{\theta \in \Theta} |J(\theta) - J_n(\theta)| \leq \frac{1}{2} \sup_{\theta \in \Theta} |P_{\theta}(H^*_\theta) - P^n_{\theta}(H^*_\theta)| + \frac{1}{2} \sup_{H \in \mathcal{H}^*} |P_{\theta^0}(H) - P^n_{\theta^0}(H)| \quad (25)$$

$$\leq \sup_{\theta \in \Theta} \sup_{H \in \mathcal{H}^*} |P_{\theta}(H) - P^n_{\theta}(H)|. \quad (26)$$

It follows that

$$P \left( \sup_{\theta \in \Theta} |J(\theta) - J^*_n(\theta)| > \epsilon \right) \leq P \left( \sup_{\theta \in \Theta} \sup_{H \in \mathcal{H}^*} |P_{\theta}(H) - P^n_{\theta}(H)| > \epsilon \right),$$

which proves the proposition. \qed
Appendix C. Supplementary Materials and Methods

C.1. Models and reference posterior distributions

We tested our inference method on several well-known distributions. This section lists the parameters used to generate the data, 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. For the Gaussian, Bernoulli, and Poisson data, the true posterior distributions are available in closed-form and are used for reference. For the moving average and ARCH models, numerical integration was employed to obtain the reference posterior distributions. The sample average of \( n \) data points \((x_1, \ldots, x_n)\) will be denoted by \( \bar{x} \), and the sample variance by \( s_n^2 \).

\[
\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i, \quad s_n^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - \bar{x})^2. \tag{27}
\]

C.1.1 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 \( \mu_0 = 1 \) and variance \( \sigma_0 = 1 \). Inference was performed on the mean \( \mu \). In the Bayesian setting, the prior distribution of \( \mu \) was Gaussian,

\[
\mu \sim \mathcal{N}(\mu_0, \sigma_0^2), \quad p(\mu|\mu_0, \sigma_0^2) = \frac{1}{\sqrt{2\pi\sigma_0^2}} \exp\left(-\frac{(\mu - \mu_0)^2}{2\sigma_0^2}\right), \tag{28}
\]

with mean \( \mu_0 = 3 \) and variance \( \sigma_0^2 = 1 \). For Gaussian data with known variance \( \sigma_0^2 \) and a Gaussian prior on the mean \( \mu \), the posterior distribution of \( \mu \) is Gaussian with mean \( \mu_n \) and variance \( \sigma_n^2 \),

\[
\mu|X \sim \mathcal{N}(\mu_n, \sigma_n^2), \quad \mu_n = \frac{\lambda_0 \mu_0 + n \bar{x}}{\lambda_0 + n}, \quad \sigma_n^2 = \frac{1}{\lambda_0 + n}, \tag{29}
\]

see, for example, (10, Chapter 2).

**Gaussian with unknown mean and variance.** The Gaussian data were generated with mean \( \mu_0 = 3 \) and variance \( \sigma_0^2 = 4 \). Both mean \( \mu \) and variance \( \sigma \) were considered unknown. In the Bayesian setting, the prior distribution was normal-inverse-gamma,

\[
\mu|v \sim \mathcal{N} \left( \mu_0, \frac{v}{\lambda_0} \right), \quad v \sim \mathcal{G}^{-1}(\alpha_0, \beta_0), \quad p(v|\alpha_0, \beta_0) = \frac{\beta_0^{\alpha_0}}{\Gamma(\alpha_0)} v^{-\alpha_0-1} \exp\left(-\frac{\beta_0}{v}\right), \tag{30}
\]

where \( \alpha_0 \) and \( \beta_0 \) are the shape and scale parameters, respectively, and \( \Gamma(t) = \int_0^\infty u^{t-1} \exp(-u)du \) is the gamma function. 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 \( \mu_n, \lambda_n, \alpha_n, \beta_n \),

\[
\mu|v, X \sim \mathcal{N} \left( \mu_n, \frac{v}{\lambda_n} \right), \quad \mu_n = \frac{\lambda_0 \mu_0 + n \bar{x}}{\lambda_0 + n}, \quad \lambda_n = \lambda_0 + n, \tag{31}
\]

\[
v|X \sim \mathcal{G}^{-1}(\alpha_n, \beta_n), \quad \alpha_n = \alpha_0 + \frac{n}{2}, \quad \beta_n = \beta_0 + \frac{n}{2} s_n^2 + \frac{n}{2} \lambda_0 \bar{x} - \mu_0)^2). \tag{32}
\]
see, for example, (10, Chapter 3).

C.1.2 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 \( \mu \) was a beta distribution with parameters \( \alpha_0 = \beta_0 = 2 \),

\[
\mu \sim \text{Beta}(\alpha_0, \beta_0), \quad p(\mu | \alpha_0, \beta_0) = \frac{\Gamma(\alpha_0 + \beta_0)}{\Gamma(\alpha_0) \Gamma(\beta_0)} \mu^{\alpha_0-1}(1 - \mu)^{\beta_0-1},
\]

which has mean 0.5 and standard deviation 0.22. The posterior is beta with parameters \( \alpha_n, \beta_n \),

\[
\mu | X \sim \text{Beta}(\alpha_n, \beta_n), \quad \alpha_n = \alpha_0 + n \bar{x}, \quad \beta_n = \beta + n(1 - \bar{x}),
\]

see, for example, (10, Chapter 2).

C.1.3 Count data

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

\[
\lambda \sim \mathcal{G}(\alpha_0, \beta_0), \quad p(\lambda | \alpha_0, \beta_0) = \frac{\beta_0^{\alpha_0}}{\Gamma(\alpha_0)} \lambda^{\alpha_0-1} \exp \left( -\beta_0 \lambda \right).
\]

The prior distribution has mean 6, mode 4, and standard deviation 3.46. The posterior distribution is gamma with parameters \( \alpha_n, \beta_n \),

\[
\lambda | X \sim \mathcal{G}(\alpha_n, \beta_n), \quad \alpha_n = \alpha_0 + n \bar{x}, \quad \beta_n = \beta_0 + n,
\]

see, for example, (10, Chapter 2).

C.1.4 Time series

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

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

\[
x_t = \epsilon_t + \theta \epsilon_{t-1}, \quad t = 1, \ldots, T,
\]

where the \( \epsilon_t, t = 0, \ldots, T \) are independent standard normal random variables, and \( \epsilon_0 \) is unobserved. The observed data were generated with \( \theta^o = 0.3 \). For the derivation of the posterior distribution, it is helpful to write the update equation in matrix form. Let \( x_{0:T} = (x_0, \ldots, x_T) \) and \( \epsilon = (\epsilon_0, \ldots, \epsilon_T) \) 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 37 can then be written as

\[
x_{0:T} = B \epsilon,
\]

\[
B = \begin{pmatrix}
1 & 0 \\
\theta & 1 & 0 \\
& \ddots & \ddots & \ddots \\
& & \ddots & 0 \\
& & & \theta & 1
\end{pmatrix}.
\]
It follows that $x_{0:T}$ is zero mean Gaussian with covariance matrix $BB^\top$. Since $x_{0:T}$ has a Gaussian distribution, we can analytically integrate out the unobserved $x_0$. The resulting vector $x_{1:T}$ is zero mean Gaussian with tridiagonal covariance matrix $C$,

$$C = \begin{pmatrix} 1 + \theta^2 & \theta & & & \\ \theta & 1 + \theta^2 & \theta & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \theta \\ & & & \theta & 1 + \theta^2 \end{pmatrix}. \quad \quad (39)$$

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

$$p(\theta|x_{1:T}) = \frac{p(x_{1:T}|\theta)}{\int_{-1}^{1} p(x_{1:T}|\theta) d\theta}, \quad \theta \in (-1,1). \quad \quad (40)$$

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

$$x_t = \theta_1 x_{t-1} + \epsilon_t, \quad \epsilon_t = \xi_t \sqrt{0.2 + \theta_2 \epsilon_{t-1}^2}, \quad t = 1, \ldots, T, \quad x_0 = 0, \quad (41)$$

where the $\xi_t$ and $\epsilon_0$ are independent standard normal random variables. We call $\theta_1$ the mean process coefficient and $\theta_2$ the variance process coefficient. The observed data consist of the $x_t$ and we generated them with $(\theta_1^o, \theta_2^o) = (0.3, 0.7)$. For the derivation of the posterior distribution, we introduce the column vectors $\epsilon = (\epsilon_1, \ldots, \epsilon_T)$ and $x_{1:T} = (x_1, \ldots, x_T)$ which are related by a linear transformation,

$$\epsilon = Qx_{1:T}, \quad Q = \begin{pmatrix} 1 & 0 & & & \\ -\theta_1 & 1 & 0 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & 0 \\ & & & -\theta_1 & 1 \end{pmatrix}. \quad \quad (42)$$

Note that the band-diagonal matrix $Q$ depends on $\theta_1$. The determinant of $Q$ is one so that

$$p_\epsilon(\epsilon_{1:T}|\theta_1, \theta_2) = p_\epsilon(Qx_{1:T}|\theta_1, \theta_2). \quad \quad (43)$$

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

$$p_\epsilon(\epsilon|\theta_1, \theta_2) = p_1(\epsilon_1|\theta_1, \theta_2) \prod_{t=2}^{T} \frac{1}{\sqrt{2\pi(0.2 + \theta_2 \epsilon_{t-1}^2)}} \exp \left(-\frac{\epsilon_t^2}{2(0.2 + \theta_2 \epsilon_{t-1}^2)}\right), \quad (44)$$

$$39$$
where $p_1$ is the pdf of $\epsilon_1$. Since $\epsilon_0$ is a latent variable following a standard normal distribution, $p_1$ is defined via an integral,

$$
p_1(\epsilon_1|\theta_1, \theta_2) = \int \frac{1}{\sqrt{2\pi(0.2 + \theta_2\epsilon_0^2)}} \exp \left( -\frac{\epsilon_1^2}{2(0.2 + \theta_2\epsilon_0^2)} \right) \frac{1}{\sqrt{2\pi}} \exp \left( -\frac{\epsilon_0^2}{2} \right) d\epsilon_0. \quad (45)
$$

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

$$
p(\theta_1, \theta_2|x_{1:T}) = \frac{p_x(Qx_{1:T}|\theta_1, \theta_2)}{\int_{-1}^{1} \int_{0}^{1} p_x(Qx_{1:T}|\theta_1, \theta_2) d\theta_1 d\theta_2}, \quad (\theta_1, \theta_2) \in (-1, 1) \times (0, 1). \quad (46)
$$

The normalizing denominator, the posterior means and variances were computed with matlab's integral2.m.
C.2. The thresholds in classifier ABC with a sequential Monte Carlo algorithm

In the employed sequential Monte Carlo ABC algorithm, a proposed parameters $\theta'$ gets accepted if the corresponding $J_n(\theta')$ is below some threshold. We chose as thresholds 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 the third iteration.

Unlike a purely quantile-based approach, this 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 Figure 23.

Figure 23: 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.
C.3. Application in epidemiology

This section is a brief review of (24).

C.3.1 Data and model

The data used by (24) consisted of the colonization state of 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 the attendee is infected with strain $s$ at time $t$. The observed data $X$ consisted thus of a set of 29 binary matrices of size $N_k \times S$ formed by the $I_{is}^{T_k}$, $i = 1, \ldots, N_k, s = 1, \ldots, S$, at time $T_k$.

The data were modeled using a continuous-time Markov chain for the transmission dynamics within a day care center, and an observation model (24). 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,

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

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 is the probability to be infected by it when previously not infected with any strain, and the last equation, the probability 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

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

where $N$ is the total number of children attending the day care center, and $S$ the total number of strains in circulation. $\Lambda$ and $\beta$ 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. $P_s$ and $S$ were determined by an analysis of the overall distribution of the strains in the cross-sectional data (yielding $S = 33$, for $P_s$, see (24)). 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 strains $s$ is $I^t_{js}/n_j(t)$).

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 \( \beta \), the external infection parameter \( \Lambda \), and the co-infection parameter \( \theta \). Uniform priors were assumed: \( \beta \in (0, 11) \), \( \Lambda \in (0, 2) \), and \( \theta \in (0, 1) \).

C.3.2 Expert statistics and distance used in ABC

A sequential Monte Carlo implementation of approximate Bayesian computation (ABC) was used to obtain posterior distributions of the three parameters \( \theta = (\beta, \Lambda, \theta) \) (24). This involved computing the distance between the observed data \( X \) and data \( Y_\theta \) simulated with parameter \( \theta \) from the model reviewed above. 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,

1. diversity of the strains present,
2. number of different strains present,
3. proportion of infected individuals,
4. proportion of individuals with more than one strain,

and each data set was summarized 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 \( X \) and \( Y_\theta \) were computed for each summary statistic, and \( \theta \) was accepted if each of the four \( L_1 \) distances was below a certain threshold. The four thresholds were adapted during the algorithm, see (24).