MINERVA SCHOOLS AT KGI

Surrogate Inference on High-Cost Simulators

Braden Scherting

February 1, 2020

CONTENTS

1 INTRODUCTION

2 BACKGROUND

2.1 Modeling

The language and procedure described below can be summarized as follows: forward modeling defines parameters and relational properties of a system; inverse modeling determines the values of parameters corresponding to observed data conditional on the pre-defined, forward relational properties.

It should be noted that many definitions of modeling are loose or domain-specific. What follows are definitions that facilitate understanding of the remainder of the paper but are not definintive. Much of the following discussion is informed and inspired by Kabanikhin (2008).

2.1.1 Forward problems

Forward or direct problems arise naturally from a desire to explain or predict systems and phenomena around us. To that end, the solution to a forward problem is a model that describes the system or phenomenon in question as some function of parameters and inputs. Formulating the forward problem additionally entails specifying a domain and boundary conditions, features which we will assume to be implicit or automatic henceforth. In mathematical and statistical modeling, the function is commonly given by equations or systems of equations. I am careful, however, to not restrict the definition of the solution function to "equations" because a central objective of this project is to enable modeling of systems that are not best represented by equations. Tunable values in the model, which modulate the relationships between random or independent quantities without changing the governing logic of the relationships, are *parameters*, represented in the general case by θ. Parameters are generally unobserved or unobservable quantities, despite often entertaining natural interpretations. For equation-based models, coefficients and initial conditions are common parameters. A parameterized model serves to predict or explain observable quantities. Those observable quantities are *data*, $\mathscr{D}$, and include independent (explanatory) variables x and dependent variables y. I denote a model with $\mathscr{M}$, which represents one possible model from a potentially infinite space of models. Thus, models are sets of parameter-tuned relationships that define a forward map from

parameters to data, $\mathscr{M} : (\theta, x) \rightarrow y$. What is commonly referred to as "modeling" is synonymous to solving the forward problem.

An important feature of forward problems is that they are usually well-posed, meaning that there exists a unique, stable solution, where stability means that any arbitrarily small change in inputs will produce a correspondingly small change in output. This is, in part, a consequence of our choosing to model natural systems, which naturally exhibit regularization and smoothing.

2.1.2 Inverse Problems

Definitions of inverse problems exist at various levels of granularity. For our purposes, the inverse problem corresponding to a given forward model $\mathscr{M}^*$ is: *what values θ^* explain or produce $\mathscr{D}^*$ conditional on $\mathscr{M}^*$?* Inverse problems prompt us to identify a parameterization or parameterizations of our model that correspond to observations. As may be obvious, this task is contingent on the choice/specification of forward model. Thus, the terminology is perhaps misleading; from the outset, forward modeling precedes inverse modeling. Inverse modeling occurs only once assumptions about the forward model have been made.

Most inverse problems are hard because they are ill-posed. Of the conditions for well-posedness, stability is often the first violated by inverse problems. Forward problems often deal with idealized relationships that are seldom realized in data due to measurement error. For even moderately complicated models, it is common for errors to propagate through the model, compounding and interacting in complicated ways. As a result, instability with respect to measurement error is a common culprit for rendering inverse problems ill-posed.

Though inverse problems are challenging, they are necessary for constructing reasoned, expressive models of systems. Because parameters are generally unobserved quantities, modeling without performing any sort of inverse or backward inference is simply guessing-and-checking. In general, solutions to inverse problems are valuable in two ways: 1) direct knowledge about meaningful parameter values and 2) access to a parameterized model with which to reason about future or unobserved data. The first mode is explanatory; knowledge about the parameters translates directly to knowledge about the system. The second is predictive; the parameterized model can be used to predict past or future data. These modes are, of course, neither mutually exclusive nor exhaustive.

Solving inverse problems (inferring model parameters from data) is the focus of entire academic and professional disciplines and is the focus of this project.

2.2 Solving Inverse Problems

Inverse problems arise from observing data and endeavouring to learn more about the model that generated or explains them. Here I discuss the two general strategies for doing so, with an emphasis on the strategy predominantly employed in this paper: Bayesian inference. For our purposes, statistical inference is synonymous with inverse modeling: learning about parameters based on observations. The two principle approaches to solving inverse problems correspond to the two predominating approaches to statistical inference.

2.2.1 Deterministic inference

Deterministic or regularization-based methods enforce a criterion such as smoothness or fit to data. The method of least squares, for example, casts statistical inference as an optimization problem where the minimization objective is the sum of squared residuals,

$$\sum_{i=1}^{N} (y^i_{obs} - \mathscr{M}(x^i_{obs}, \theta))^2. \tag{2.1}$$

Optimization-based methods characterize a broad swath of statistical inference. The general strategy is to define a model (e.g. linear regression: $y = mx + b$), define a differentiable objective (e.g. sum of squared residuals), and employ an efficient optimum-finding algorithm. Only for simple models is the optimization convex; modern, flexible models often force us to settle for locally optimal solutions. Optimization-based statistical inference or maximum likelihood estimation (MLE) plays an important, supporting role in this project.

Countless regularization criteria can be employed to solve inverse problems deterministically. A major concern with these approaches, however, is how to select the type and strength of regularization. In machine learning, the choice of regularization is is synonymous with crafting a loss function. Cross-validation and hyperparameter tuning are strategies for empirically determining the strength of regularization. In any case, the result of a regularization-based inverse problem solving strategy is a point estimate of the solution parameters.

2.2.2 Probabilistic inference

The primary statistical inference perspective interrogated in this project is Bayesian inference. Regularization- and optimization-based inverse problem solving methods commonly incorporate probability. However, Bayesian inference formaly regards the unknown parameter values as beloning to probability distributions. Thus, the governing theorem of Bayesian inference (Bayes' theorem or Bayes' rule) is derived directly from statements of probability theory. Before presenting the brief derivation of this rule, I'd like to contemplate Bayesian inference from a general perspective.

A popular interpretation of the theorem is as an *update*. Whereas deterministic methods introduce assumptions via regularizers, Bayesian methods introduce assumptions via probabilistic expressions of prior beliefs. Any analysis begins with a preconceived belief about the values of parameters (even if that belief is very uncertain). The belief may follow from previous analyses or from fundamental constraints (e.g. populations are positive). Upon observing some new data, we would like neither to replace old beliefs nor disregard the new data. Rather, we would like to incorporate new evidence into existing beliefs, thereby arriving at an updated expression for the belief. See chapter 1 of McElreath (2018) for a description of Bayesian analysis devoid of references to beliefs or subjectivity.

Bayesian inference is a formal approach to probabilistically solving ill-posed inverse problems. The strategy targets the conditional probability of parameters given data $p(\theta|y)$. Let's now examine Bayes' rule:

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)} \tag{2.2}$$

Working backward to arrive at this statement, consider two statements of conditional probability from the product rule of probability (we introduce A and B as any two arbitrary quantities):

$$p(A|B) = \frac{p(A,B)}{p(B)} \tag{2.3}$$

$$p(B|A) = \frac{p(B,A)}{p(A)} \tag{2.4}$$

Notice that joint probabilities are symmetric: $p(B, A) = p(A, B)$. Thus, (2.4) can be reexpressed and rearranged as

$$p(B|A) = \frac{p(A, B)}{p(A)} \tag{2.5}$$

$$p(A, B) = p(B|A)p(A). \tag{2.6}$$

We now have a statement that can be substituted into (2.3), arriving at Bayes' rule:

$$p(A|B) = \frac{p(B|A)p(A)}{p(B)} \tag{2.7}$$

There are numerous ways to read and interpret the four terms of Bayes' rule. I present each in turn along with interpretations that facilitate the subsequent discussion. Upon developing familiarity with the terms, I proceed to a discussion of Bayes' theorem holistically and Bayesian inference.

PRIOR: $p(\theta)$ Prior–a fitting name for the term representing prior beliefs. Bayes' theorem requires that we formally, probabilistically express existing beliefs about the parameter values in the prior. This is a probability distribution over parameters, and, in non-hierarchical models, it is a marginal probability. If the system has never before been investigated and there is no information about possible values for the parameters, the prior should be constructed to express this uncertainty (i.e., uninformative prior). Alternatively, if the system has been extensively researched, the prior should convey the conclusions of the past research (i.e., informative prior). Because the prior necessarily entails subjectivity, it is a feature of Bayesian inference that often receives criticism. And, indeed, inappropriately crafting a prior distribution can lead to severely misleading results. However, the prior is, in large part, the defining feature of Bayesian inference and enables conclusive inference on underconstrained problems.

NORMALIZING CONSTANT: $p(y)$ Despite being a constant, this term is a source of unending angst. The normalizing constant (or marginal probability of data or model evidence) gives the marginal probability of the observed data conditioned only on the model and can be best un-

derstood from its definition:

$$p(y) = \int p(y|\theta)p(\theta)d\theta \tag{2.8}$$

This term answers the question, What is the probability of the data, under the model but irrespective of the parameter values? A practical interpretation of this term is that it serves to normalize the invalid probability distribution expressed by the product in the numerator, thereby ensuring that the posterior is a valid probability distribution. This interpretation implies the first, alternative formulation of Bayes' theorem we will encounter:

$$p(y|\theta) \propto p(y|\theta)p(\theta) \tag{2.9}$$

The angst that results from this term is due to the intractability of the integral in (2.8). The main thrust of classical Bayesian inference methods is to approximate or otherwise work around the evaluation of this integral.

LIKELIHOOD: $p(y|\theta)$ OR $\mathscr{L}(\theta|y)$ The likelihood is the term responsible for incorporating the evidence or data according to the model. It is also responsible for significant confusion: the likelihood function *is not a valid probability distribution.* It is represented as a conditional probability $p(\cdot|\cdot)$, but it is not a valid probability. Consider what constitutes a valid probability distribution: a function that assigns probabilities to values of random variables, the total probability of which is exactly 1. With likelihood functions, trouble arises from how we define the random variable. In Bayesian inference, we have observed y; the data is not random. θ is unobserved, so despite the likelihood describing the probability of y given θ, $\mathscr{L}$ is a function of θ: $\mathscr{L}(\theta)$. $p(y|\theta)$ for some fixed theta and unknown y would be a valid probability distribution. Consider the probability density function of the normal distribution:

$$\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.10}$$

It is intentionally left undefined. The familiar normal distribution is defined in terms of the random variable x, $\mathcal{N}(x|\mu,\sigma)$:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.11}$$

A normal likelihood expresses our belief that the data is distributed normally given pa-

rameters $\theta = \{\mu, \sigma\}$. The data is known and fixed. The parameters, however, are unknown, resulting in a function of θ:

$$f(\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.12}$$

Defining the function in terms of θ bereaves us of a normal distribution. The functional form is the same, but the variable designation is different. Recall that the total probability a random variable taking on a value is 1 (i.e. the function integrates to 1):

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \, dx = 1 \tag{2.13}$$

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \, d\theta \neq 1 \tag{2.14}$$

A likelihood function returns the *likelihood* that particular parameters θ correspond to y, a value analogous to probability but not probability.

Importantly, the way that $\mathscr{L}(\theta)$ incorporates data reflects the model. The normal likelihood in (2.12) assumes the data is symmetric and normally distributed around a mean μ. If, however, data is, for example, non-negative, such a model may be inappropriate. The likelihood function should express the forward model of the system.

POSTERIOR: $p(\theta|y)$ This is the quantity of interest: the distribution over parameters conditional on observed data. It provides estimates of parameter values *and* the associated uncertainty. Once a posterior has been computed, it represents the fully updated belief of parameter values. Thus, upon observing new data, it can be used as the prior distribution.

2.2.3 A Note on Hierarchical Modeling

When faced with data that has natural, grouped structure, such as cohorts of patients from different hospitals or words in sentences, a statistician ignorant of hierarchical or multilevel modeling has two options: perform independent analyses for each group (cohort, sentence) and apply post hoc averaging, or pool all observations (patients, words) into a single group and perform analysis for the entire population. At one extreme, each group is assumed to be independent of the others. At the other, there is assumed to be no group-specific information.

Hierarchical modeling blends these approaches by treating each group as a population within a meta-population. Formally, a hierarchical setup models group-level parameters as random variables by placing hyperpriors $p(\theta|\phi)$ over the parameters and inferring the associated hyperparameters ϕ,

$$p(\phi, \theta|y) = \frac{p(y|\theta)\,p(\theta|\phi)\,p(\phi)}{p(y)} \tag{2.15}$$

A hierarchical model allows a researcher to ask questions about unobserved units within an observed group or about an unobserved group. The values of inferred hyperparameters also indicate the similarity among groups.

Hierarchichal modeling is arguably the most important development in Bayesian statistics in recent decades, with some authors (e.g., Gelman et al. (2013), McElreath (2018)) suggesting that it be the default approach to many inference tasks.

2.2.4 Bayesian methods for probabilistic inference

Here, I will briefly introduce strategies for computing posterior distributions. Many of the claims I make about each class of methods refer to the general case; for example, while sampling methods are *generally* slower and more accurate than variational methods, there are exceptions in both directions. Take each claim with a grain of salt. See Gelman et al. (2013) for more comprehensive and in-depth coverage of these topics.

MAP Though it does not represent a fully-Bayesian treatment of a problem, maximum a posteriori (MAP) estimation is nonetheless a mainstay of Bayesian methods. Related to maximum likelihood estimation, MAP seeks the maximum of the posterior distribution, following from the fact that,

$$\begin{aligned} \arg\max_{\theta} p(\theta|y) &= \arg\max_{\theta} \frac{p(y|\theta)\,p(\theta)}{p(y)} \\ &= \arg\max_{\theta} p(y|\theta)\,p(\theta), \end{aligned} \tag{2.16}$$

because the normalizing constant does not depend on θ. This has the advantage of obviating the evaluation of the integral in the numerator but reduces the posterior to a point estimate, thereby discarding many of the advantages confered by Bayesian inference.

CONJUGATE DISTRIBUTIONS: *exact, analytical* Though we often dismiss the integral in the denominator of Bayes' theorem as intractable, there are cases for which it is analytically, exactly evaluable. For trivial problems or problems with small, discrete data/parameter space (where the integral reduces to a sum), this is obviously true. However, there are particular combinations of well-known probability distributions that integrate nicely. For likelihoods belonging to standard distributional families, there usually exists what is known as a conjugate prior distribution. Evaluating Bayes' rule (including the integral) gives a posterior distribution belonging to the same family as the conjugate prior. The hyperparameters (parameters of the posterior distribution) can thus be updated according to a deterministic update rule informed by data. Conjugate priors are generally not solutions to arbitrary problems in Bayesian inference as they necessarily constrain the choice of prior. However, they are both an important fundamental, fully-Bayesian method and constitute the basis for some of the most expressive, performant Bayesian methods around (e.g., Gaussian processes).

DISTRIBUTIONAL APPROXIMATIONS: *approximate, analytical* Distributional and modal approximations attempt to approximate the posterior distribution or the respective factors of the posterior with tractable distributions. This often entails formulating an optimization problem. Modal approximations are exemplified by the Laplace approximation, which applies a Gaussian approximation to the posterior by matching the mean with the maximum of the unnormalized posterior and computing an approximation to the variance by use of second derivative information (Laplace's method). The accuracy of the approximation plummets for non-normal posteriors (e.g., multi-modality, skewness, etc.) (MacKay, 2003). It can be, however, startlingly fast to compute. Variational methods can be used to approximate the posterior by identifying an approximating distribution or distributional family and minimizing the Kullback-Leibler (KL) divergence between the approximating distribution and the posterior. Convergence and accuracy of variational methods depends on, among other things, the validity of approximating family and optimization method. Improvements to these features constitute improvements to the method. For example, stochastic variational inference improves on classical variational inference by replacing the expectation-maximization/gradient ascent algorithm with stochastic optimization (Hoffman et al., 2013). Expectation propagation (EP) (Minka, 2001) assumes a factorization of the multidimensional posterior and iteratively optimizes each factor. EP is thus often the slowest but most accurate of these methods. All of the methods here require ac-

cess to the unnormalized posterior density (product of prior and likelihood). For example, the objective of many variational methods, the evidence lower bound,

$$\mathrm{E}_q \left[\log \left(\frac{p(\theta, y)}{q(\theta)} \right) \right], \tag{2.17}$$

where $q(\cdot)$ is the approximating distribution, can be seen to contain the numerator of Bayes' rule inside of the logarithm.

SAMPLING METHODS: *exact*, numerical* The strategy of sampling-based methods is to generate samples from the target distribution i.e., $p(\theta|y)$. The breadth of these methods is staggering, but they adhere to principles of Monte Carlo sampling (transforming random numbers into quantities of interest). Monte Carlo methods that sample from Markov Chains, known as Markov Chain Monte Carlo (MCMC) simulation, also inherit convergence properties from Markov theory. The outcomes of these methods are sets of uncorrelated or correlated samples from the posterior distribution and are thereby non-parametric. Sample statistics, with sufficiently many samples and a reasonable sampling scheme, correspond to statistics of the target distribution. Upon convergence, sampling methods produce samples from the posterior distribution. Sample statistics of these samples are exact in the limit of infinite samples (*) and thus represent some of the most accurate Bayesian methods. However, they also rank among the slowest methods. As with all other classes of methods discusses here, they require access to the unnormalized posterior.

In summary, Bayesian inference is a principled framework for solving inverse-problems. However, it is non-trivial task because computing the posterior analytically or exactly is often impossible. Importantly, the classical methods that exist for posterior computation are all reliant on access to the unnormalized posterior density.

2.3 Likelihood-Free Inference

Performing standard probabilistic inference by use of Bayes' rule requires access to the likelihood of the data in addition to the forward model. This entails expressing the foward model as a statement of probability. When researchers anticipate using Bayesian inference to formulate their inverse problem, it is common to simply define the forward model probabilistically from

the outset. Other models, which define a map from parameter and input space to data space, can be straightforwardly written as a likelihood. However, a large class of models evades such treatment, *likelihood-free models*. Membership in this class is somewhat contrived in that any model for which the likelihood is intractable or unavailable is a likelihood-free model. In practice, these are commonly referred to as mechanistic, rule-based, or simulator-based models. This nomenclature follows from the fact that simulators, while highly expressive, often possess only an implicit likelihood; samples can be generated from the model by running the simulator for a set of parameters and inputs according to well-understood, natural mechanisms, but the likelihood of any outcome relative to others is unknown. An alternative way to understand the intractability of the likelihood is by the definition of the likelihood,

$$p(y|\theta) = \int p(y, z|\theta) dz, \tag{2.18}$$

where z are latent variables related to the data-generating process; the nature of the latent space varies greatly between models. Probabilistic models define $p(y|\theta)$ directly, thus obviating the integral in (2.18). The upshot is that for likelihood-free models, the integral in (2.18) is intractable (in addition to (2.8)). In the absence of an evaluable likelihood, standard Bayesian inference methods fall flat.

Enter likelihood-free inference (LFI). Those familiar with classical literature, may be more accustomed to the term approximate Bayesian computation (ABC). The distinction between the two is not set in stone, and they largely refer to the same process. In this paper, I use LFI as a general term and reserve ABC for exclusively sampling-based methods. That being said, LFI is a collection of methods dedicated to enabling probabilistic inference on simulator-based models.

Why is this task important? Often times, describing how data arises or how systems evolve is much more natural and justifiable than constructing a probabilistic model from the outset. This is the motivation for modeling strategies like agent-based modeling and differential equations. Much of the original development of ABC methods was motivated by the size and complexity of data in population genetics. LFI is thus closely related to advancing science: it enables statistical inference on descriptive, mechanistic physical models.

Generally, LFI methods approximate the posterior distribution over parameters by comparing forward simulations to observed data. Consider the Rejection-ABC scheme (Pritchard

et al., 1999), which generates N samples from the approximate posterior distribution:

1. For i in $1:N$
2. **Do:**
3. $\theta_* \sim p(\theta)$
4. $y_{\theta_*} \sim \mathscr{M}(\theta_*)$
5. **Until:** $\Delta(y_{obs}, y_{\theta_*}) \leq \epsilon$
6. $\theta_i = \theta_*$
7. $\Theta = \{\Theta, \theta_i\}$

In prose, sample parameters from the prior; simulate data for the sampled parameters; if the simulated data is epsilon-close to the observed data, add the data-generating parameters to the set of posterior samples. Pritchard et al.'s innovations were sampling from a prior and introducing an accept/reject tolerance (ϵ). The result is an approximation to the posterior distribution $p_\epsilon(\theta|y)$. In the limit of $\epsilon \to 0$, this quantity converges to the true posterior $p(\theta|y)$. Note also, however, that being a sampling-based method implies that the posterior is only correct in the limit of $n \to \infty$. Thus, $p_\epsilon(\theta|y)$ will approach the sample posterior with small ϵ. It can straightforwardly seen that ϵ is a trade-off parameter; the smaller the tolerance, the fewer samples are accepted and the more simulations need to be run to reach N posterior samples. Small values of ϵ quickly become computationally intractable.

For high-dimensional y, computing the discrepancy Δ becomes increasingly difficult. Beaumont et al. (2002) propose the use of summary statistics $S(y)$ in place of uncompressed data, yielding a posterior $p_\epsilon(\theta|S)$. A posterior of this form is equivalent to $p_\epsilon(\theta|y)$ if and only if $S(\cdot)$ is statistically sufficient (lossless compression). The use of summary statistics, however, enables the comparison of rich data formats, including high-dimensional and time series data, given an appropriate statistic(s). This ability is generally assumed to outweigh the bias induced by use of a reduced representation. Choice of S is non-trivial; designing and choosing S is usually left to domain-experts. In recent years, there has been development in the area of learned summary statistics i.e. a neural network learns an optimal, reduced representation of data.

A suite of natural extensions to rejection-ABC emanating from the Monte Carlo and MCMC literature have been applied to LFI problems. The Metropolis-Hastings algorithm can be applied to ABC by slightly modifying the acceptance probability of a transition from x to candidate x^*:

$$A(x^*|x) = \mathbb{I}\left[\Delta(y, y_0) \leq \epsilon\right] \mathbf{min}\left(1, \frac{p(\theta)}{p(\theta^*)} \frac{q(\theta|\theta^*)}{q(\theta^*|\theta)}\right) \tag{2.19}$$

where $x = (\theta, y)$, and q is a proposal distribution that depends only on θ. Replacing the rejection step in the above algorithm with this Metropolis-Hastings step yields the MCMC-ABC algorithm described by Marjoram et al. (2003). It has the advantage of preventing the need to naïvely sample from the prior. To further address the problem of tolerance setting, a number of papers propose iterative algorithms that reduce the tolerance with each iteration (e.g. Del Moral et al. (2012), Sisson et al. (2007), Beaumont et al. (2009)). Collectively, these are known as sequential Monte Carlo-ABC (SMC-ABC).

Another important class of methods, first introduced by Wood (2010), attempts to approximate the unavailable, implicit likelihood. Wood uses a Gaussian approximation to the distribution of *summary statistics*. This follows from the central limit theorem and is principled as $n \to \infty$. In possession of a Gaussian approximation to the likelihood, you can proceed with standard Bayesian inference. However, when n is finite, the Gaussianity assumption becomes restrictive. Non-Gaussian synthetic likelihood approaches have become a key feature of modern LFI.

2.3.1 Density Estimation

For a set of samples $X = \{x_1 \dots x_N\}$ drawn from an arbitrary, unknown probability density p, the task of density estimation is identify an estimate $\hat{p}$ of the true density. Density estimation is the foundation of numerous statistical and machine learning methods because it bridges the gap between discrete data and distributions that can be used in arbitrary analyses. Given this role, density estimators must be evaluable, valid probability distributions. Evaluable estimators can be sampled from via MCMC methods, but it is often valuable for density estimators to be directly sampleable.

Density estimation and its neural network, variant neural density estimation, are increasingly important features of LFI. The method of synthetic likelihood can be straightforwardly generalized by replacing the learned Gaussian approximation with a flexible density estimator. This is the strategy taken by Alsing et al. (2019) and others. A good approximation to the likelihood can be used in place of or in conjunction with the simulator to enhance inference. We take a similar approach in this project. Our density estimation method of choice is real-valued

non-volume preserving transformations (real NVP) (Dinh et al., 2017).

REAL NVP Recent developments in density estimation and probabilistic generative modeling (e.g. Goodfellow et al. (2014), Rezende et al. (2014)) employ approximate inference or auxiliary models, such as discriminatory networks, to enforce learning conditions and arrive at a learned density. Real NVP is an approach to density estimation that takes a much simpler approach and requires neither approximate inference nor auxiliary models. Most approaches to generative modeling involve learning a latent representation of the data space, a latent space. Some feature of the latent space (e.g. dimension, functional form) is specified and the inference step entails learning the transformation between the data space and the latent space. Dinh et al. propose defining the transformations as invertible bijections, thereby allowing for the use of change of variable formula. The transformations are thus defined in terms of a scaling function s and translation t. A single-layer map from data to latent space is thus,

$$z = \exp(s(x) + t(x)). \tag{2.20}$$

Crucially, the method also involves layering these transformations, only transforming a subset of dimensions in each layer (masking). They layering enables flexibility, the masking enforces a non-linearity that prevents the sqeuential linear transformations from reducing to a single linear transformation. The change of variable formula requires evaluating the Jacobian of this transformation. The clever construction of this transformation, however, yields a Jacobian that does not involve the Jacobian of s or t. Therefore, s and t can be arbitrarily complicated functions, so the natural step is to make them deep neural networks, which can be trained using maximum likelihood. In this way, Real NVP produces a learned transformation from the N-dimensional data space to an N-dimensional unit Gaussian, which is both evaluable and sampleable.

2.3.2 Bayesian Optimization for Likelihood-Free Inference

Bayesian optimization for likelihood-free inference (BOLFI) is the LFI method central to the workflow presented here. I will first introduce Bayesian optimization and the active learning paradigm and then explain its use in LFI. Broadly, under favorable conditions, BOLFI is the most sample-efficient LFI method in widespread use.

BAYESIAN OPTIMIZATION Bayesian optimization can be succinctly described as derivative-free, global optimization of black-box functions. "Derivative-free" refers to the fact that we neither have access to closed-from derivative information nor resort to automatic differentiation (autodiff), thus precluding methods like gradient descent. Similarly, black-box functions are not available in closed-form and can only be queried/evaluated. This is analogous to likelihood-free simulators. Bayesian optimization proceeds by first approximating the objective function f^* within a Bayesian paradigm. The most natural way to do this is by use of a Gaussian process (GP) surrogate. A prior distribution $p(f)$ over the space of possible functions f is combined with queried data of the form $\mathscr{D} = \{x_{1:N}, f^*(x_{1:N})\}$ via a [tractable] likelihood $p(\mathscr{D}|f)$, yielding a posterior distribution over functions: $p(f|\mathscr{D})$. We arrive at this posterior via an analytical, conjugate update (for more, see the distinguished Rasmussen and Williams (2005)). This posterior can be understood as a function which is exact for $x \in \mathscr{D}$ (assuming a noiseless model) and which approximates and quantifies uncertainty for $x \notin \mathscr{D}$. More precisely, the posterior defines Gaussian distributions at every value of X with means and variances informed by $\mathscr{D}$. Naïvely sampling the function, fitting the GP surrogate to samples, and computing the minimum (or maximum) of the surrogate (which is possible due to the properties GPs) would seem to solve the problem. The nuance and strength of Bayesian optimization, however, is the active learning step baked in. Active learning (also known as experimental design) places control over where to sample next in the hands of the algorithm. The uncertainty quantification intrinsic to GPs can be leveraged to identify regions of the input space likely to contain optima. The identification of these regions is left to *acquisition functions*. Acquisition functions are mini optimization problems that take as input the current posterior and output a value x^* that maximizes an acquisition objective. These functions must balance exploration (targeting areas of high posterior variance) and exploitation (targeting areas of low/high posterior mean). While evaluating acquisition functions is sometimes costly, the increase in sample-efficiency conferred usually greatly outweighs the cost. Brochu et al. (2010) offer a comprehensive review of Bayesian optimization methods.

BOLFI Features of Bayesian optimization seem eminently applicable to LFI: probabilistic modeling, tractable posteriors, sample-efficiency. Gutmann et al. (2016) propose minimizing the discrepancy function Δ. Data is thus of the form $\mathscr{D} = \{\theta_{1:N}, \Delta(y_{obs}, \mathscr{M}(\theta_{1:N}))\}$. BOLFI thus proceeds by actively selecting parameters from the prior that are likely to minimize the distance

between observed data and simulated data. Thresholding the GP regression a small amount above the global minimum and sampling from the truncated Gaussian distributions produces samples from the approximate posterior. BOLFI is an extremely sample-efficient method, consistently producing high-fidelity posteriors with $\mathcal{O}(N^2)$ simulations. Many of the primary drawbacks of the method (evidence ceiling, dimensionality ceiling, tuning) are inherited from the regression model: GP. Because full GP updating necessitates inverting the $N \times N$ covariance matrix, an operation in $\mathcal{O}(N^3)$, the number of samples is limited to the order of thousands. Nonetheless, BOLFI and variants exist as the only tenable LFI methods for high-cost simulators.

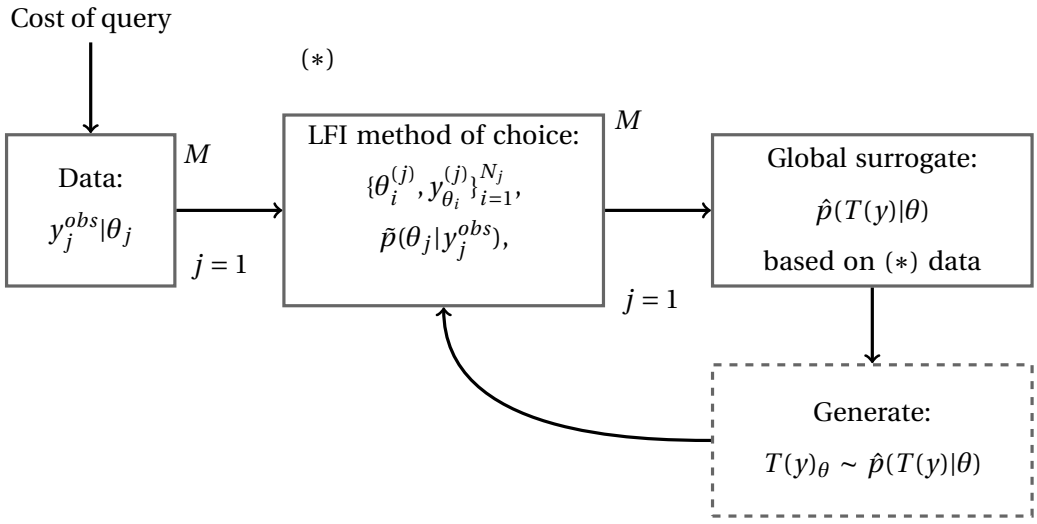
2.3.3 Engine for Likelihood-Free Inference

Engine for likelihood-free inference (ELFI) (Lintusaari et al., 2018) is a probabilistic program and Python package for LFI. It includes, among other things, implementations of rejection-ABC, SMC-ABC, BOLFI, and a number of simulators. This package serves as the basis for much of the LFI computation presented in this project. ELFI is by no means the only LFI package. However, it is an appropriate choice because it 1) is under active development, 2) implements the necessary LFI methods, and 3) has featured in high-impact journal articles (e.g. in Nature Microbiology, Shen et al. (2019)).

3 METHODS: WORKFLOW

The main contribution of this project is the introduction and validation of a general workflow for multi-observation likelihood-free inference on high-cost simulators by use of fast, global surrogate likelihood. The workflow, in its entirety is presented first, followed by motivation and explanation of each component of the workflow. While being a general workflow, I attempt to offer grounding by way of examples for each component. I conclude by synthesizing the workflow and considering limitations, alternative interpretations, and possible extensions.

Full workflow:



For observations $j = 1 : M$ corresponding to unique sets of parameters of interest, the workflow proceeds as follows:

1. Compute crude, initial posterior updates for each observation:

$$\{\tilde{p}(\theta_j|y_j^{obs}) \mid j = 1 \dots M\}$$

2. Pool N evidence from each of the M local posteriors:

$$\{\{(\theta_i^{(j)}, y_{\theta_i}^{(j)}) \mid i = 1 \dots N\} \mid j = 1 \dots M\} \Rightarrow \{(\theta_k, y_{\theta_k}) \mid k = 1 \dots M \times N\}$$

3. Train/learn approximation to global likelihood (surrogate likelihood) of summary statistics:

$$\hat{p}(T(y)|\theta)$$

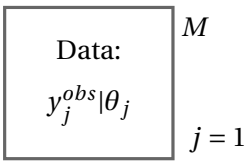
4. Replace expensive simulator with surrogate likelihood simulator:

$$T(y)_\theta \sim \hat{p}(T(y)|\theta)$$

5. Proceed with local, posterior inference.

3.1 Motivation

3.1.1 Multi-observation LFI



In the standard LFI setting, the target is $p(\theta|y_{obs})$, where y_{obs} is a single realization of a data generating process. A complete "fit" of an LFI method, be it via MCMC, density estimation, or Bayesian optimization, returns a posterior belief about the values of parameters that probably generated the data, according to the model. If we collect multiple observations, each with a unique set of parameters of interest but which are assumed to come from the same model, standard LFI methods must be run multiple times or in parallel to recover all posteriors. Samples may be reused across solutions, but for all but the simplest methods (e.g. rejection-ABC), acquisition of candidate parameter-data is largely localized to areas of high posterior probability for the respective solutions, meaning that $\Delta > \epsilon$ is likely for most shared samples. Additionally, processes that share samples are no longer embarrassingly parallel. This workflow endeavours to share information among local posteriors to recover a global representation of the likelihood that can be used to aid in local inference for each of the posteriors and enable near-immediate inference on subsequent observations.

Thus, the first step in the workflow is observation of M data. One dataum may be as complicated as a snippet of time-series recordings, samples from a process, or the state of a system after a fixed amount of time. In any case each datum corresponds to a single parameter setting.

3.1.2 Expensive simulators

If the simulator in question is fast (order of seconds), this workflow is not relevant. Parallel or even sequential implementations are sufficient for high-quality inference in reasonable time. However, when a single simulation takes minutes or hours to run, and there are multiple, relevant posteriors of interest, LFI quickly becomes infeasible. Sophisticated MCMC-ABC methods may require hundreds of thousands of simulations per observation. This workflow is most rele-

vant for tasks where the simulation budget is less than 10,000 simulations. BOLFI (and variants) is the only method I am aware of that is capable of computing a posterior with $< O(10^3)$ simulations. For complicated, realistic simulators, computing even a single posterior may take several weeks in a high-performance computing environment. Enabling inference on high-cost simulators of this complexity enables science.

This work presented here exists in parallel to and in support of a forthcoming publication authored by Henri Pesonen, Braden Scherting, and Samuel Kaski. Though the strategy has been hinted at, for example by Alsing et al. (2019),

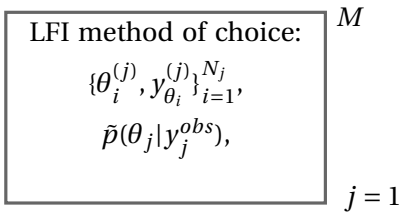
> However, in some scenarios we may run many "experiments" that generate independent realizations of data d from the same data generating process, and we want to analyze those data as they are taken. In these situations, it is desirable to abandon active learning and build a global emulator for p($\mathbf{d}|\theta$) over the full prior volume, that can then be used to analyze any subsequent data **d** as they are observed.

there is not been, to my knowledge, an end-to-end investigation. This workflow directly addresses two of the three shortcomings of contemporary LFI described by Cranmer et al. (2019): amortization and sample efficiency. The third shortcoming, quality of inference, while of critical importance, is not the main objective of the workflow. The workflow aims to *enable* inference where it was formerly impossible.

3.1.3 Hierarchical LFI

- (Tran et al., 2017)
- (Bazin et al., 2010)

3.2 Incomplete Local Posteriors / Crude Fits



As noted above, BOLFI is the LFI method of choice for the analyses performed here because is a natural choice in high-cost simulation settings. That being said, the workflow is not restricted to BOFLI-based inference. Upon observing multiple data, it is necessary to query

the simulator to gather information about the likelihood space, even in this sample-efficient scheme. To do this, we opt to grant control of evidence querying to the local problems. In practice, this means initiating LFI for each observation in parallel. Each local problem will query the simulator for N locally optimal evidence, such that $N \times M$ is less than or equal to our simulation budget. If we have chosen a problem for which this method is justified, the local posteriors that result will be crude approximations to full, converged posteriors. $N \times M$ evidence points may be sufficient to fully solve a single or even multiple posteriors, but are insufficient to solve all M posteriors. However, by performing initial LFI on *all* local posteriors, the evidence points should cover the likelihood space better than if all evidence were actively aquired by a single posterior (i.e. M weak local optima vs 1 strong local optimum).

3.3 Global Likelihood Surrogate

Global surrogate:
$\hat{p}(T(y)|\theta)$
based on $(*)$ data

In possession of $N \times M$ evidence points from across the parameter space but which slightly prefer the respective local optima, we can approximate the conditional distribution that describes the data: the global likelihood $p(y|\theta)$. Because LFI posteriors are fit in terms of summary statistics and because raw data is often unwieldy, we change the target distribution slightly to $p(T(y)|\theta)$, the likelihood of summary statistics. This quantity differs from many synthetic likelihood LFI methods in that the learned likelihood should be able to describe arbitrary observations, each with different parameters. For density estimation purposes, this quantity corresponds to a conditional (on θ) density estimate. Thus, an appropriate density estimation method should return an estimator that can be evaluated for a pair (θ, T) and yield samples $T_\theta \sim \hat{p}(T|\theta)$. This learned, global likelihood can then be regarded as a surrogate simulator with an evaluable likelihood.

3.3.1 Surrogate substrate

How to model the surrogate simulator or global likelihood ranked among the more important questions that arose in developing this workflow. Consider the role that the surrogate plays.

The surrogate pools information from multiple local posterior inferences, each with their own locally optimal acquisitions (i.e. simulator queries). The pooled acquisitions are used to learn a global representation of the likelihood, which can itself be sampled in place of the expensive simulator. This role implies the necessary traits of a satisfactory surrogate substrate:

- Generative (sampleable)
- Sufficiently expressive
- Inexpensive to train

A surrogate must be samplable in order to stand-in for the simulator. Because likelihood-free models are typically black-boxes, we are unable to constrain the complexity of the implicit likelihood and thus use a surrogate that is able to express highly abnormal distributions. Lastly, because we are operating in a high-cost computing environment (i.e. expensive simulator), learning the surrogate must be data-efficient.

In validating this workflow, I experimented with a number of surrogate models. The original motivation was to build on recent developments in using neural density estimation for likelihood free inference. The pydelfi (density estimation likelihood-free inference) package (Alsing et al., 2019) provides implementations of a number of NDEs (e.g. mixture density networks, masked autoencoder for density estimation, etc.) but results with these implementations were unsatisfactory. I simplified the surrogate substrate and instead assumed independence of the summary statistics and modeled each with a GP. This was simply to validate the method. I found this to perform well on simple task, thus indicating the ability of the workflow. As the complexity and dimension of the simulator grew, however, this approach became worse, as was expected. Having found evidence of the method, I searched for a more flexible method that could more straightforwardly handle multidimensional output. This was satisfied by real NVP.

TRAINING THE SURROGATE The real NVP surrogate is trained via maximum likelihood learning. The loss function used is the negative log probability of data under the latent distribution. That is, the log probability is evaluated by first transforming data to the latent space and then evaluating the latent probability of the transformed data. This loss functions encourages the surrogate to assign greater probability to data-dense regions. I use batch-wise, stochastic gradient descent with a decaying learning rate and track the loss on a held-out validation set. Train-

ing terminates when the maximum number of epochs has been reached or the validation loss fails to improve for 20 consecutive epochs, at which point, the parameter values revert to the minimal training loss setting. Included in the attached Python file (`cnvp.py`) is also a simpler training scheme which optimizes the network parameters for a fixed number of epochs.

3.4 Simulator/Surrogate swap

Generate:
$T(y)_\theta \sim \hat{p}(T(y)|\theta)$

A learned, global likelihood that can be straightforwardly sampled from has a natural interpretation as a simulation. By fixing the settings θ and simulating (sampling) a random variable the conditioned distribution, we obtain a realization of the simulator for the given settings. Samples from a perfect surrogate $p(T(y)|\theta)$ would be indistinguishable from samples from the true simulator. However, sampling from the surrogate is significantly faster (given an appropriate choice of surrogate model). Our surrogate is likely not to be perfect but can, with some caveats, be assumed to be a good approximation to the likelihood implicit in the expensive simulator. Thus, we exchange accuracy for efficiency. Each of the local LFI fits are granted access to the surrogate simulator as if it is the true simulator and proceed with inference with an effectively unlimited simulation budget.

3.5 Methods discussion

4 Experiments

I test the ability of global surrogate likelihoods to accelerate or enable multi-observation inference on two problems familiar to statistics literature and a third, formerly intractable, applied problem.

If successful, this method will *recover reference solutions* with 5-10x fewer simulations, thus enabling inference in high-cost domains where reference solutions are unavailable. Because the surrogates are intended to be low-cost, high-fidelity replacements for simulators,

non-identifiability or other flaws in reference solutions are expected and preferred to be reflected in surrogate solutions.

4.1 g-and-k

Whereas most familiar models in statistics are defined in terms of their likelihoods, quantile distributions are defined in terms their quantile functions or inverse cumulative distribution functions (inverse CDF). Because the inverse CDF is assumed to be known in closed-form, quantile distributions are trivial to sample from via inverse transform sampling, but corresponding likelihoods are commonly unavailable. The g-and-k distribution is a five-parameter quantile distribution defined in terms of the standard normal quantile function, $z(p)$:

$$\begin{aligned} Q_{gk}\left(p \mid A, B, g, k\right) = A + B\left(1 + c\frac{1-\exp(-g\,z(p)}{1+\exp(-g\,z(p)}\right) \\ \times \left(1+z(p)^2\right)^k z(p) \end{aligned} \tag{4.1}$$

A controls location, B scale, g skewness, and k kurtosis. By convention, c, an asymmetry-related parameter, is fixed at 0.8 (Allingham et al., 2009). The absence of a tractable likelihood and the interpretability of the parameters make the g-and-k distribution a natual choice for LFI experimentation.

I adopt the simulation study setup of Drovandi and Pettitt (2011). Uniform priors on A, B, g, and k with respective ranges $(0,5)$, $(0,5)$, $(-5,5)$, $(-0.5,5)$ are used. Observations are each represented by 10000 samples from a uniquely parameterized g-and-k distribution and are summarized by robust estimates of the first four moments, as proposed by Drovandi and Pettitt. As they discuss, different regions of the parameters space are more or less amenable to different numbers of samples and different summary statistics. This setup is chosen because it performs well across the parameter space, rather than optimally for individual parameters of interest.

To test the multi-observation framework, I first sampled 10 sets of true parameters from the uniform priors. I then simulated one observation (10000 samples) from each of the resulting, parameterized g-and-k distributions to represent independent observations. For each observation, I fit an instance of BOLFI with 35 points, 20 of which were initial, random evidence and 15 of which were actively chosen, optimal evidence. The dotted lines in 4.1 denote

the posterior distributions obtained by use of the 35 acquisition points alone. I then collected all evidence from all observations (350 points) and learned a real NVP surrogate likelihood. In posession of the surrogate, I replaced the g-and-k simulator with the sampleable, surrogate model and continued with BOLFI inference until reaching 250 acquisitions. The posteriors resulting from this are shown in red. Solid black posteriors correspond to the target, BOLFI posteriors trained with 250 evidence points, all of which are acquired from the g-and-k simulator.

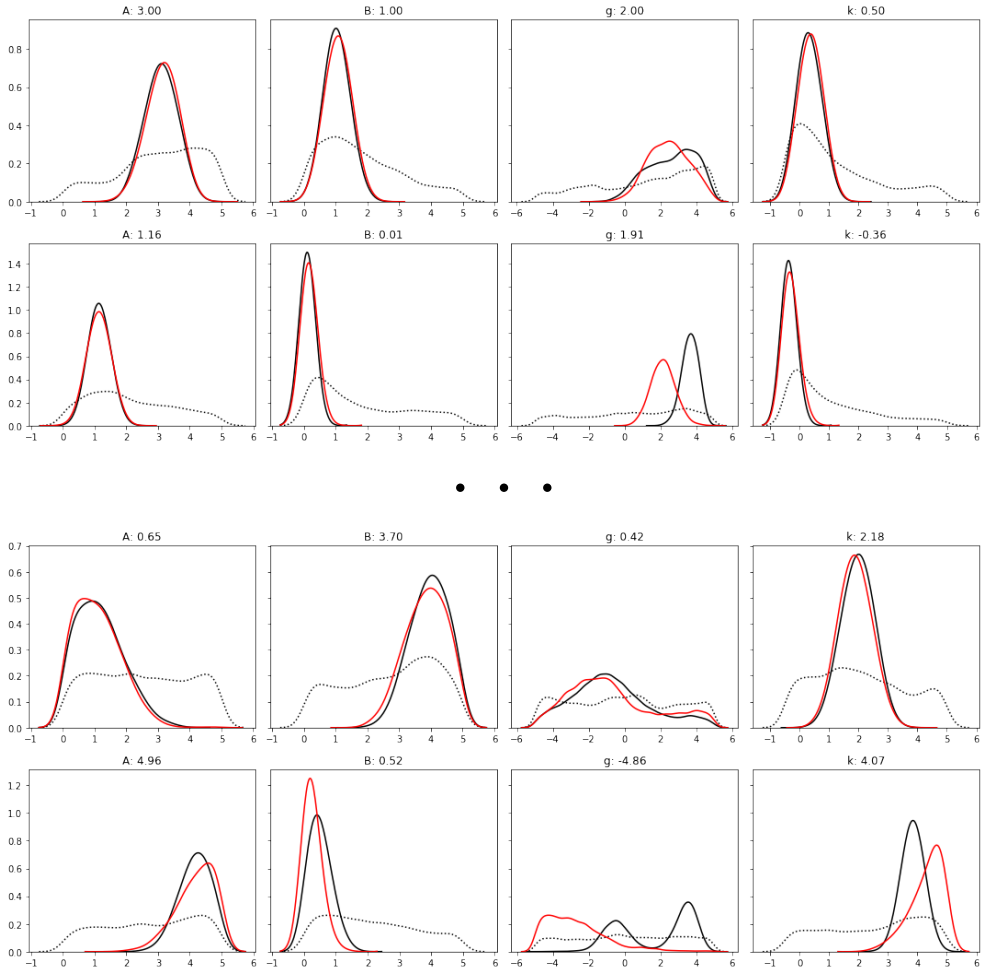


Figure 4.1: Marginal posteriors for each of the four parameters (A, B, g, and k) for four randomly selected observations (out of 10 total).

ANALYSIS The surrogate method can be seen to recover the full-BOLFI solutions with high fidelity. When the surrogate solution differs from the full-BOLFI solution, the surrogate solution is generally biased in the direction of the true parameter value. Perhaps the most encouraging feature of these results is that the surrogate solutions exhibit roughly the same variance as

the reference solutions; uncertainty in reference solutions is reflected in surrogate solutions. It would be concerning if surrogate solutions displayed high-confidence or convergent behavior when the reference solution did not. Results of this nature are consistent across replications. I find these results to be both impressive and compelling.

4.2 Next Simulator

5 Discussion

6 Conclusion

REFERENCES

Allingham, D., King, R. A. R., and Mengersen, K. L. (2009). Bayesian estimation of quantile distributions. *Statistics and Computing*, 19(2):189–201.

Alsing, J., Charnock, T., Feeney, S., and Wandelt, B. (2019). Fast likelihood-free cosmology with neural density estimators and active learning. *Monthly Notices of the Royal Astronomical Society*.

Bazin, E., Dawson, K. J., and Beaumont, M. A. (2010). Likelihood-free inference of population structure and local adaptation in a bayesian hierarchical model. *Genetics*, 185(2):587–602.

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

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

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

Cranmer, K., Brehmer, J., and Louppe, G. (2019). The frontier of simulation-based inference.

Del Moral, P., Doucet, A., and Jasra, A. (2012). An adaptive sequential monte carlo method for approximate bayesian computation. *Statistics and Computing*, 22(5):1009–1020.

Dinh, L., Sohl-Dickstein, J., and Bengio, S. (2017). Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Drovandi, C. C. and Pettitt, A. N. (2011). Likelihood-free bayesian estimation of multivariate quantile distributions. *Computational Statistics and Data Analysis*, 55(9):2541–2556.

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian data analysis*. Chapman and Hall/CRC.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C.,

Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.

Gutmann, M. U., Corander, J., et al. (2016). Bayesian optimization for likelihood-free inference of simulator-based statistical models. *Journal of Machine Learning Research*.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347.

Kabanikhin, S. I. (2008). Definitions and examples of inverse and ill-posed problems. *Journal of Inverse and Ill-Posed Problems*, 16(4):317–357.

Lintusaari, J., Vuollekoski, H., Kangasrääsiö, A., Skytén, K., Järvenpää, M., Marttinen, P., Gutmann, M. U., Vehtari, A., Corander, J., and Kaski, S. (2018). Elfi: Engine for likelihood-free inference. *Journal of Machine Learning Research*, 19(16):1–7.

MacKay, D. (2003). Information theory, pattern recognition and neural networks.

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

McElreath, R. (2018). *Statistical rethinking: A Bayesian course with examples in R and Stan*. Chapman and Hall/CRC.

Minka, T. P. (2001). Expectation propagation for approximate bayesian inference. In *Proceedings of the Seventeenth conference on Uncertainty in artificial intelligence*, pages 362–369. Morgan Kaufmann Publishers Inc.

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

Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. In Xing, E. P. and Jebara, T., editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1278–1286, Bejing, China. PMLR.

Shen, P., Lees, J. A., Bee, G. C. W., Brown, S. P., and Weiser, J. N. (2019). Pneumococcal quorum sensing drives an asymmetric owner–intruder competitive strategy during carriage via the competence regulon. *Nature microbiology*, 4(1):198.

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

Tran, D., Ranganath, R., and Blei, D. (2017). Hierarchical implicit models and likelihood-free variational inference. In *Advances in Neural Information Processing Systems*, pages 5523–5533.

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