UiO : Department of Informatics
University of Oslo

Computational Science: Imaging and Biomedical Computing

Master Essay

Likelihood-free Inference Methods for Parameter Identification in Mechanistic Models

---

Nicolai Haug

June 1, 2020

github.com/nicolossus/Master-thesis

Contents

1 Introduction **1**

2 Background **2**
2.1 Notation . . . 2
2.2 Bayesian Inference . . . 2
2.3 Likelihood-free Inference . . . 4
2.4 Approximate Bayesian Computation . . . 5
2.4.1 Rejection ABC . . . 5
2.4.2 Markov-chain Monte Carlo ABC . . . 6
2.5 Approximation of Intractable Likelihoods . . . 7
2.5.1 Synthetic Likelihood . . . 7

3 Recent Methodologies **7**
3.1 Sequential Neural Posterior Estimation . . . 8
3.2 Sequential Neural Likelihood . . . 9

4 Learning the Posterior vs. the Likelihood **9**

5 Outlook **10**

References **12**

1 Introduction

Mechanistic models in neuroscience aim to explain neural or behavioral phenomena in terms of causal mechanisms, and candidate models are validated by investigating whether proposed mechanisms can explain how experimental data manifests. A central challenge in building a mechanistic model is to identify the parametrization of the system which achieves an agreement between the model and experimental data. Finding well-fitted parameters by inspection becomes more difficult as the complexity of both data and models increase, and automated identification of data-compatible parameters become necessary [1].

Statistical inference provides the mathematical means and procedures for automated parameter identification, and is usually the method of choice. Statistical inference uses the likelihood $p(\boldsymbol{x} \mid \boldsymbol{\theta})$ to quantify the match between parameters $\boldsymbol{\theta}$ and data $\boldsymbol{x}$ by deriving estimators of $\boldsymbol{\theta}$ from the data. In Bayesian inference, posterior beliefs about parameters $\boldsymbol{\theta}$ given data $\boldsymbol{x}$, $p(\boldsymbol{\theta} \mid \boldsymbol{x})$, can be obtained by multiplying the likelihood $p(\boldsymbol{x} \mid \boldsymbol{\theta})$ with prior beliefs $p(\boldsymbol{\theta})$ and normalizing [2]. However, many mechanistic models are defined implicitly through *simulators*, i.e. a set of dynamical equations and possibly a description of sources of stochasticity, which can be run forward to generate data [3]. Likelihoods can be derived for purely statistical models, but are generally intractable or computationally infeasible for simulation-based models [1]. Hence are traditional methods in the toolkit of statistical inference inaccessible for many mechanistic models. To overcome this, a suite of methods that bypass the evaluation of the likelihood function, called *likelihood-free inference* methods, have been developed. These methods seek to directly estimate either the posterior or the likelihood, and require only the ability to generate data from the simulator to analyze the model in a fully Bayesian context [2].

Approximate Bayesian Computation (ABC) constitutes a class of computational methods rooted in Bayesian statistics that can be used to evaluate posterior distributions of model parameters without having to explicitly calculate likelihoods [4]. At its heart, the ABC approach is quite simple. Evaluation of the likelihood is replaced by comparing synthetic data (generated by the model) to observed data, in order to assess how likely it is the model could have produced the observed data. In this essay, two types of ABC methods will be discussed: the vanilla *rejection ABC* and the more sophisticated variant *Markov chain Monte Carlo ABC*.

ABC methods seek to directly estimate the posterior, but viable alternative methods instead seek to estimate the intractable likelihood function for the model of interest. Synthetic Likelihood (SL), which aim to improve computational efficiency relative to ABC, uses a multivariate normal approximation to the summary statistic likelihood. This auxiliary likelihood can be incorporated within a Bayesian framework, which is referred to as BSL [5, p. 322].

Recently, there have been several successful studies using neural network-based conditional density estimators to perform likelihood-free inference in simulation-based models.

Sequential Neural Posterior Estimation (SNPE) target parametrically learning the posterior by using simulations instead of likelihood calculations. Instead of filtering out simulations, as ABC methods do, it uses *all* simulations to train the neural network to identify admissible parameters [3].

Sequential Neural Likelihood (SNL) learns a parametrized surrogate likelihood, meaning that it requires additional inference procedures to compute the posterior. However, learning

the likelihood can be advantageous, as it is often easier to learn compared to the posterior. Furthermore, a model of the likelihood can be reused with different priors [2].

The goal of this essay is to provide an overview of likelihood-free inference and its recent advancements in the context of parameter identification in mechanistic models. As there are methods for both estimating the posterior and estimating the likelihood, a natural question to explore is which approach is preferable.

This essay is structured by first presenting a theoretical overview of Bayesian inference and established likelihood-free inference methods in Section 2. This is followed by a presentation of the recent methodologies SNPE and SNL in Section 3. Next, a comparison of the likelihood-free inference methods are given in Section 4. Lastly, an outline for possible continuations for the master project is presented in Section 5.

2 Background

2.1 Notation

At the outset, a few comments on notation: First, $p(\cdot \mid \cdot)$ denotes a *conditional probability density* and $p(\cdot)$ denotes a *marginal distribution*. The conditional probability $P(A \mid B)$ is the likelihood of event A occurring given that B is true, and the marginal probability $P(A)$ is the probability of observing A. The terms *distribution* and *density* are used interchangeably. For brevity, the term *probability density* will often be condensed into the term *density*. A *probability mass function*, that gives the probability that a discrete random variable is exactly equal to some value, is abbreviated *pmf*. Similarly, a *probability density function*, associated with continuous rather than discrete random variables, is abbreviated *pdf*. The same notation is used for continuous density functions and discrete probability mass functions.

2.2 Bayesian Inference

In statistical inference, there are, broadly speaking, two paradigms for the analysis of sampled data: frequentist inference and Bayesian inference. They often differ with each other in their fundamental interpretation of probability. In terms of parameter inference, the standard frequentist view is to regard the value of some parameter θ as fixed but unknown, and draw appropriate conclusions from sample data $\boldsymbol{x}$ by emphasizing the relative frequency of the data. The Bayesian approach to inference is to regard θ as a random variable having a *prior probability distribution*, usually a prior pdf $p(\theta)$ since there will typically be a continuum of possible values of a parameter rather than just a discrete set, that incorporates all available information about it [6, p. 758, 776]. In the case of substantial prior knowledge about a parameter θ, the prior pdf is narrow and concentrated about some central value, whereas a lack of information yield a wider and relatively flat prior pdf as shown in Figure 2.1.

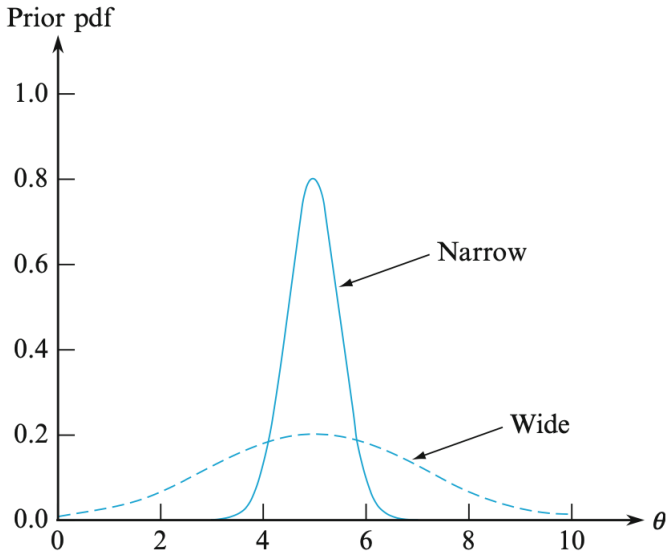

Figure 2.1: *A narrow concentrated prior about some central value and a wider less informative prior. Retrieved from [6, Figure 14.3].*

In order to make probability statements about θ given sample data $\boldsymbol{x}$, a probabilistic model representing the joint probability distribution for θ and $\boldsymbol{x}$ must be provided [7, p. 6]. The joint pmf or pdf can be written as a product of the prior distribution $p(\theta)$ and the conditional *sampling distribution* $p(\boldsymbol{x} \mid \theta)$:

$$p(\theta, \boldsymbol{x}) = p(\theta)p(\boldsymbol{x} \mid \theta)$$

At this point, Bayes' theorem is used to produce the *posterior* distribution of θ given the data $\boldsymbol{x}$ [6, p. 758, 776]. A common incarnation of Bayes' theorem is

$$p(\theta \mid \boldsymbol{x}) = \frac{p(\theta, \boldsymbol{x})}{p(\boldsymbol{x})} = \frac{p(\boldsymbol{x} \mid \theta)p(\theta)}{p(\boldsymbol{x})} \tag{2.1}$$

As may be evident from above, the various terms in Bayes' theorem have formal names. For clarity, these terms will now be discussed in further detail. The quantity $p(\theta)$ is called the *prior* (probability) distribution and represents beliefs about θ before the data $\boldsymbol{x}$ is analyzed. It is often specified by a particular distribution among a set of well-known and tractable distributions, with the purpose of making evaluation of prior probabilities and random generation of θ values straightforward [4]. For instance, if the inferential problem concerns the height of Scandinavian women, one might expect an average height around 165 cm and a Gaussian centered at that specific height is a reasonable prior. In the case of lack of prior information, a uniform prior can be used. Our prior state of knowledge is modified by data $\boldsymbol{x}$, obtained by performing experiments, through the *likelihood* function $p(\boldsymbol{x} \mid \theta)$, and yields the *posterior* distribution $p(\theta \mid \boldsymbol{x})$, representing our state of knowledge about θ in the light of $\boldsymbol{x}$ [8, p. 6]. The mean of this posterior distribution gives a point estimate of θ. An interval having a posterior probability .95 gives a 95% *credibility* interval, an interval within which an unobserved parameter value falls with a particular probability, the Bayesian analogue of a 95% confidence interval [6, p. 777]. The quantity $p(\boldsymbol{x})$ is called the *evidence* (sometimes also termed the marginal likelihood or the prior predictive probability of the

data), and is the distribution of the observed data marginalized over the parameter [4]. In the case of continuous θ the evidence is given by $p(\boldsymbol{x}) = \int p(\theta)p(\boldsymbol{x} \mid \theta)\mathrm{d}\theta$, and in the case of discrete set of parameters by $p(\boldsymbol{x}) = \sum_\theta p(\theta)p(\boldsymbol{x} \mid \theta)$, where the sum is over all possible values of θ [7, p. 7]. The evidence is the same for all possible θ, as it does not depend on θ, meaning that, with fixed $\boldsymbol{x}$, this factor can be omitted in parameter identification since it constitutes a normalizing constant and does not enter into determining the relative posterior probabilities of different values of θ [4]. Omitting the evidence yield the unnormalized posterior distribution:

$$p(\theta \mid \boldsymbol{x}) \propto p(\theta, \boldsymbol{x}) = p(\boldsymbol{x} \mid \theta)p(\theta) \tag{2.2}$$

The second term in this expression, $p(\boldsymbol{x} \mid \theta)$, is taken here as a function of θ, not of $\boldsymbol{x}$.

The core of Bayesian inference is encapsulated in Equation 2.1 and Equation 2.2. The principal task is to develop the joint probability model $p(\theta, \boldsymbol{x})$ and perform the computations to summarize the posterior $p(\theta \mid \boldsymbol{x})$.

2.3 Likelihood-free Inference

Suppose a data-generating process is controlled by parameters $\boldsymbol{\theta}$. When the process is run forward it stochastically generates a datapoint $\boldsymbol{x}$ whose distribution depends on $\boldsymbol{\theta}$. For every setting of $\boldsymbol{\theta}$, assume that the process defines a conditional density function $p(\boldsymbol{x} \mid \boldsymbol{\theta})$. Given an observed datapoint $\boldsymbol{x}_0$ known to be generated by the process, the problem of interest is inferring plausible parameter settings that could have generated $\boldsymbol{x}_0$. In particular, computing the posterior density $p(\boldsymbol{\theta} \mid \boldsymbol{x} = \boldsymbol{x}_0)$ obtained by Bayes theorem (Equation 2.1) is of interest. The choice of inference algorithm primarily depends on how the data-generating process is modelled [9, p. 54].

A purely statistical model, also known as a *density model* or *explicit model*, describes the conditional density function $p(\boldsymbol{x} \mid \boldsymbol{\theta})$ of the process given values for $\boldsymbol{x}$ and $\boldsymbol{\theta}$. With a density model, the posterior density $p(\boldsymbol{\theta} \mid \boldsymbol{x} = \boldsymbol{x}_0)$ is, in general, easily evaluated using Bayes theorem. Even though the normalizing constant (or evidence) $p(\boldsymbol{x}_0)$ is typically intractable, samples from the posterior can be generated using a number of popular algorithms such as importance sampling and Markov chain Monte Carlo, or the posterior can be approximated with a more convenient distribution using e.g. variational inference. Such methods are referred to as *likelihood-based inference methods*, as they explicitly evaluate the likelihood $p(\boldsymbol{x} \mid \boldsymbol{\theta})$ [9, p. 55] [5, p. 4].

On the contrary, a *simulator model*, also known as an *implicit model*, describes how the process generates data. Many mechanical models are implicitly defined through simulator models, that is, as a set of dynamical equations and possibly a description of stochastic processes. For any parameter setting $\boldsymbol{\theta}$, a simulator model can be run forward to generate independent samples from $p(\boldsymbol{x} \mid \boldsymbol{\theta})$. Unlike for explicit density models, likelihoods are generally intractable or computationally infeasible for complex data-generating processes such as simulation-based models. The absence or complexity of the associated likelihood typically arise from it involving computationally expensive or intractable integrals, or that the simulator's internal states are unavailable. In order to perform inference in a simulator model, methods using simulations from the model rather than likelihood evaluations are needed. Such methods are referred to as *likelihood-free inference methods* [2] [3] [9, p. 55].

In general, likelihood-free methods are less efficient than likelihood-based methods as the former can require lots of simulations to produce accurate results. One of the principal topics of research in likelihood-free inference is how to obtain state-of-the-art results with fewer simulations [10].

2.4 Approximate Bayesian Computation

Approximate Bayesian Computation (ABC) constitutes a class of computational methods rooted in Bayesian statistics that can be used to evaluate posterior distributions of model parameters without having to explicitly calculate likelihoods. ABC methods approximate the likelihood function by assessing how likely it is the model could have produced the observed data, based on comparing synthetic data generated by the simulator to the observed data. The simulations that do not reproduce the observed data within a specified tolerance are discarded [4] [5, p. ix].

ABC methods have been successfully applied to a wide range of real-world problems, and have also paved the way for a range of other likelihood-free approaches. However, even though ABC methods are mathematically well-founded, they inevitably make assumptions and approximations whose impact needs to be carefully assessed [4]. In the following, three types of ABC methods will be discussed: the vanilla *rejection ABC*, and the more sophisticated variant *Markov chain Monte Carlo (MCMC) ABC*.

2.4.1 Rejection ABC

Given observed data $\boldsymbol{x}_0$ and synthetic data $\boldsymbol{x}$ generated by a simulator, let $\rho(\cdot, \cdot)$ be a distance metric (e.g., the Euclidean norm) defined in data space $\mathbb{R}^D$ and $\epsilon \geq 0$ be a tolerance. For small ϵ, the ABC approximation to the posterior is

$$p(\boldsymbol{\theta} \mid \boldsymbol{x} = \boldsymbol{x}_0) \simeq p(\boldsymbol{\theta} \mid \rho(\boldsymbol{x}, \boldsymbol{x}_0) \leq \epsilon) \tag{2.3}$$

Rejection ABC is a rejection-sampling method for obtaining independent samples from the approximate posterior $p(\boldsymbol{\theta} \mid \rho(\boldsymbol{x}, \boldsymbol{x}_0) \leq \epsilon)$. It works by first sampling a set of parameters from the prior $p(\boldsymbol{\theta})$, then simulating data under the model specified by the sampled parameters, and only accepting and retaining the sample if the distance between $\boldsymbol{x}$ and $\boldsymbol{x}_0$ is no more than ϵ. The tolerance parameter ϵ controls the trade-off between estimation accuracy and computational efficiency. With sufficiently small ϵ, and a sensible distance metric, the accepted samples follow the exact posterior more closely, though the algorithm accepts less often. On the other hand, the algorithm accepts more often with a large ϵ, but the accepted samples will yield a replica of the prior [9, p. 58] [5].

An issue with ABC in general is that the required number of simulations increases dramatically as ϵ becomes small. Moreover, likelihood-free inference also becomes challenging when the dimensionality of the data is large. A common approach to lessen this problem is to use lower-dimensional summary statistics, $S(\boldsymbol{x})$ and $S(\boldsymbol{x}_0)$, that capture important features such as the mean and standard deviation, in place of raw data [2]. A further motivation for this approach is that real-world experiments often are interested in capturing summary statistics of the experimental data. A summary statistic that contains the same amount of information about model parameters as the whole dataset, is referred to as being a *sufficient statistic* [4]. The acceptance criterion in the rejection ABC algorithm then becomes:

$$\rho(S(\boldsymbol{x}), S(\boldsymbol{x}_0)) \tag{2.4}$$

In Figure 2.2, a conceptual overview of the rejection ABC algorithm is shown.

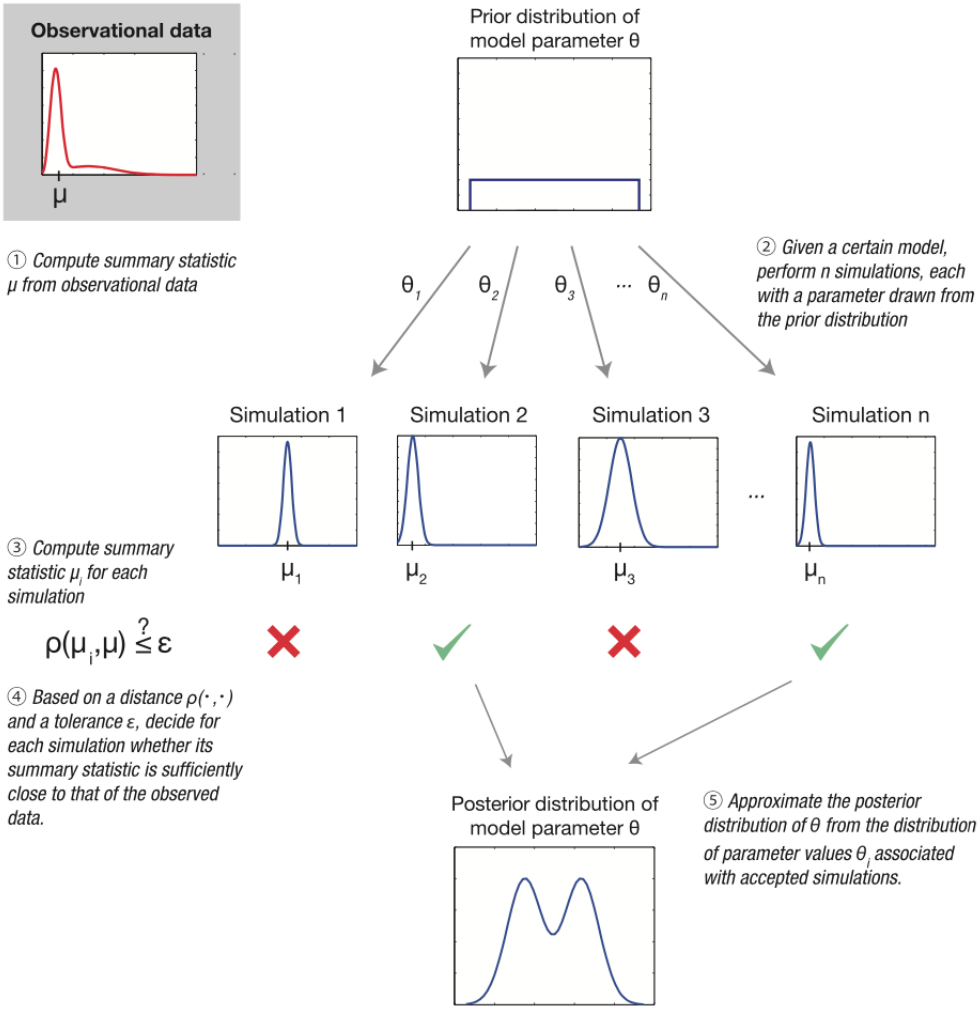

Figure 2.2: *Parameter estimation by Approximate Bayesian Computation: a conceptual overview. Retrieved from [4, Figure 1].*

2.4.2 Markov-chain Monte Carlo ABC

In the rejection ABC algorithm, parameters are sample from the prior $p(\boldsymbol{\theta})$, and only parameters that are likely under the approximate posterior $p(\boldsymbol{\theta} \mid \rho(\boldsymbol{x}, \boldsymbol{x}_0) \leq \epsilon)$ are accepted. The acceptance rate will be low if the approximate posterior is significantly narrower than the prior, as is often the case [9, p. 59].

Markov-chain Monte Carlo (MCMC) ABC is an alternative approach that can lead to fewer rejections. Instead of proposing parameters from the prior, this method use the Metropolis-Hastings algorithm to propose new parameters $\boldsymbol{\theta}'$ based on previously accepted parameters $\boldsymbol{\theta}$ from the proposal density $q(\boldsymbol{\theta}' \mid \boldsymbol{\theta})$. By calculating the *acceptance ratio*

$$\alpha = \frac{p(\rho(\boldsymbol{x}, \boldsymbol{x}_0) \leq \epsilon \mid \boldsymbol{\theta}')p(\boldsymbol{\theta}')q(\boldsymbol{\theta} \mid \boldsymbol{\theta}')}{p(\rho(\boldsymbol{x}, \boldsymbol{x}_0) \leq \epsilon \mid \boldsymbol{\theta})p(\boldsymbol{\theta})q(\boldsymbol{\theta}' \mid \boldsymbol{\theta})}, \tag{2.5}$$

the algorithm outputs the proposed parameters $\boldsymbol{\theta}'$ with probability $\min(1, \alpha)$, otherwise it outputs the previous parameters $\boldsymbol{\theta}$ [9, p. 59].

The approximate likelihood $p(\rho(\boldsymbol{x}, \boldsymbol{x}_0) \leq \epsilon \mid \boldsymbol{\theta})$ cannot be directly evaluated in the likelihood-free situation, but it can be estimated as the fraction of the simulated data $\boldsymbol{x}$ whose distance from the observed data $\boldsymbol{x}_0$ is no more than ϵ:

$$p(\rho(\boldsymbol{x}, \boldsymbol{x}_0) \leq \epsilon \mid \boldsymbol{\theta}) \approx \frac{1}{N} \sum_n I(\rho(\boldsymbol{x}_n, \boldsymbol{x}_0) \leq \epsilon), \tag{2.6}$$

where $\boldsymbol{x}_n \sim p(\boldsymbol{x} \mid \boldsymbol{\theta})$ and $I(\cdot)$ is an indicator function [9, p. 59].

Similarly to rejection ABC, the acceptance probability of MCMC ABC decreases as ϵ becomes small. Moreover, the performance of MCMC ABC strongly depends on the selection of proposal and prior density.

2.5 Approximation of Intractable Likelihoods

ABC methods seek to directly estimate the posterior, but suffers from a curse of dimensionality of the summary statistic. Since the ABC likelihood is in effect based on kernel estimation of the summary statistic density, high-dimensional summary statistics become challenging. There are several viable methods that instead seek to estimate the intractable likelihood, with the aim to improve computational efficiency relative to ABC. One such method is *Synthetic Likelihood*.

2.5.1 Synthetic Likelihood

Synthetic Likelihood (SL) uses a multivariate normal density with mean μ and covariance matrix Σ, that can depend on the parameters, to approximate the summary statistic likelihood. For simulated $S_1, ..., S_N \sim p(S(\boldsymbol{x} \mid \theta)$, the mean and covariance matrix are computed according to [11]:

$$\mu(\boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} S_i, \tag{2.7}$$

$$\Sigma(\boldsymbol{\theta}) = \frac{1}{N-1} \sum_{i=1}^{N} (S_i - \mu(\boldsymbol{\theta}))(S_i - \mu(\boldsymbol{\theta}))^T \tag{2.8}$$

This is in turn used as the likelihood $\phi(S(\boldsymbol{x}_0); \mu(\boldsymbol{\theta}), \Sigma(\boldsymbol{\theta}))$, where $\phi(z; \mu, \Sigma)$ is the multivariate normal. This auxiliary likelihood can be incorporated within a Bayesian framework, which is referred to as BSL. The BSL approach requires substantially less tuning than ABC. It also becomes increasingly more computationally efficient than ABC with an increase in the dimension of the summary statistic due to the parametric approximation of the summary statistic likelihood. However, the BSL approach remains simulation intensive and strong departures from normality can lead to poor approximations [5, p. 322].

3 Recent Methodologies

Recently, there have been several successful studies using neural network-based conditional density estimators to perform likelihood-free inference in simulation-based models. In the

following, two novel methods is presented; *Sequential Neural Posterior Estimation (SNPE)* and *Sequential Neural Likelihood (SNL)*.

3.1 Sequential Neural Posterior Estimation

Sequential Neural Posterior Estimation (SNPE)[1] is a novel method for parameter inference. The method uses ABC to learn a neural network which maps features of observed data to the posterior distribution over parameters. The strategy was originally proposed by Papamakarios and Murray in [12] and further developed by Lueckmann et al. in [3] and [1]. In the literature, the variant of Papamakarios and Murray is often referred to as SNPE-A and the variant of Lueckmann et al. as SNPE-B. In this essay, SNPE refer to the particular method by Lueckmann et al.

A *conditional neural density estimator* is a parametric density model $q_{\boldsymbol{\phi}}$ (such as a neural network), where $\boldsymbol{\phi}$ are distribution parameters. With a pair of datapoints $(\boldsymbol{u}, \boldsymbol{v})$ as input, the model outputs a conditional probability density $q_{\boldsymbol{\phi}}(\boldsymbol{u} \mid \boldsymbol{v})$. Given a set of training data $\{\boldsymbol{u}_n, \boldsymbol{v}_n\}_{1:\,N}$ that are independent and identically distributed according to a joint probability density $p(\boldsymbol{u}, \boldsymbol{v})$, $q_{\boldsymbol{\phi}}$ is trained by minimizing the loss $\mathcal{L} = -\sum_n \log q_{\boldsymbol{\phi}}(\boldsymbol{u}_n, \boldsymbol{v}_n)$ with respect to $\boldsymbol{\phi}$. With enough training data, and with a sufficiently flexible model, $q_{\boldsymbol{\phi}}(\boldsymbol{u} \mid \boldsymbol{v})$ will learn to approximate the conditional $p(\boldsymbol{u} \mid \boldsymbol{v})$ [2].

A neural density estimator $q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid \boldsymbol{x})$ can be used to approximate the posterior $p(\boldsymbol{\theta} \mid \boldsymbol{x}_0)$ as follows. First, a set of samples $\{\boldsymbol{\theta}_n, \boldsymbol{x}_n\}_{1:\,N}$ is obtained from the joint distribution $p(\boldsymbol{\theta}, \boldsymbol{x})$, by $\boldsymbol{\theta}_n \sim p(\boldsymbol{\theta})$ and $\boldsymbol{x}_n \sim p(\boldsymbol{x} \mid \boldsymbol{\theta}_n)$ for $n = 1, ..., N$. Then, $q_{\boldsymbol{\phi}}$ is trained using $\{\boldsymbol{\theta}_n, \boldsymbol{x}_n\}_{1:\,N}$ as training data in order to a global approximation of $p(\boldsymbol{\theta} \mid \boldsymbol{x})$. Finally, $p(\boldsymbol{\theta} \mid \boldsymbol{x}_0)$ can be simply estimated by $q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid \boldsymbol{x}_0)$. In order to obtain an accurate posterior fit, this approach may require a large number of simulations to sample enough training data in the vicinity of $\boldsymbol{x}_0$ [2].

SNPE is a strategy for reducing the number of simulations needed by conditional neural estimation. Since simulations from parameters with low posterior density $p(\boldsymbol{\theta} \mid \boldsymbol{x}_0)$ may not be useful in training $q_{\boldsymbol{\phi}}$, the key idea of SNPE is to generate parameter samples $\boldsymbol{\theta}_n$ from a proposal $\tilde{p}(\boldsymbol{\theta})$, that generates data $\boldsymbol{x}_n$ more likely to be in the vicinity of $\boldsymbol{x}_0$, instead of the prior $p(\boldsymbol{\theta})$ [2]. However, minimizing $\mathcal{L}$ on samples drawn from a proposal $\tilde{p}(\boldsymbol{\theta})$ no longer yields the target posterior but rather the *proposal posterior*

$$\tilde{p}(\boldsymbol{\theta} \mid \boldsymbol{x}) = p(\boldsymbol{\theta} \mid \boldsymbol{x}) \frac{\tilde{p}(\boldsymbol{\theta}) p(\boldsymbol{x})}{p(\boldsymbol{\theta}) \tilde{p}(\boldsymbol{x})}, \tag{3.1}$$

where $\tilde{p}(\boldsymbol{x}) = \int \tilde{p}(\boldsymbol{\theta}) p(\boldsymbol{x} \mid \boldsymbol{\theta}) d\boldsymbol{\theta}$ and it is assumed that $\tilde{p}(\boldsymbol{\theta}) = 0$ where $p(\boldsymbol{\theta}) = 0$ [13]. Hence, to account for sampling from a proposal $\tilde{p}(\boldsymbol{\theta})$, an adjustment of either the learned posterior or the proposed samples must be made.

Lueckmann et al. deal with this problem by adjusting the parameter samples $\boldsymbol{\theta}_n$ by assigning them weights $w_n = p(\boldsymbol{\theta}_n)/\tilde{p}(\boldsymbol{\theta}_n)$. During training SNPE minimizes an importance weighted loss $-\sum_n w_n \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_n \mid \boldsymbol{x}_n)$, which allows for direct recovery of $p(\boldsymbol{\theta} \mid \boldsymbol{x})$ from $q_{\boldsymbol{\phi}}$ with no correction and no restrictions on $p(\boldsymbol{\theta})$, $\tilde{p}(\boldsymbol{\theta})$ or $q_{\boldsymbol{\phi}}$ [13]. However, the weights can have a high variance, which can lead to slow or inaccurate inference [2].

In Figure 3.1, a conceptual overview of the SNPE method is shown.

[1] Source code available at https://github.com/mackelab/delfi

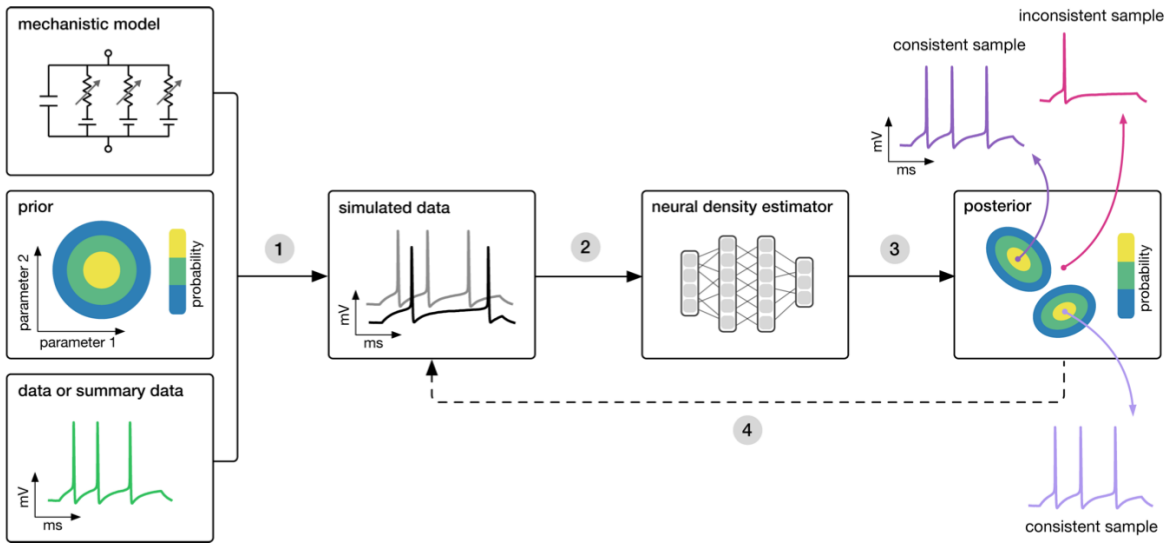

Figure 3.1: *Parameter estimation by Sequential Neural Posterior Estimation (SNPE): a conceptual overview. Retrieved from [1, Figure 1].*

3.2 Sequential Neural Likelihood

Sequential Neural Likelihood (SNL)[2] is a novel method method for Bayesian inference by Papamakarios et al. presented in [2].

SNL learns a neural conditional density estimate of the likelihood, $q_\phi(\boldsymbol{x}_0 \mid \boldsymbol{\theta}) \approx p(\boldsymbol{x}_0 \mid \boldsymbol{\theta})$, and thus avoids the bias introduced by the proposal in SNPE. SNL also does not need corrections and restrictions on $p(\boldsymbol{\theta})$, $\tilde{p}(\boldsymbol{\theta})$ or q_ϕ, allowing for choices, suitable for the task at hand, that results in learning a model of the likelihood in the region of high posterior density. Since SNL learns a parametrized surrogate likelihood, additional inference procedures are needed to compute the posterior. However, learning the likelihood can be advantageous, as it is often easier to learn compared to the posterior. Furthermore, a model of the likelihood can be reused with different priors [2]. In some cases, it can be more difficult to estimate the likelihood rather than the posterior [13].

4 Learning the Posterior vs. the Likelihood

A general question is whether it is preferable to learn the posterior or the likelihood. ABC and SNPE learn the posterior, whereas (B)SL and SNL target the likelihood. In the studies presenting SNPE, [3][1], and SNL, [2], it is shown that these neural network-based methods outperform the traditional methods. One might argue that to understand how a model quantitatively explains data, it is necessary to find not only the *best*, but *all* parameter settings consistent with experimental data. Both SNPE and SNL use simulations instead of likelihood calculations, but instead of filtering out simulations, they use *all* simulations to train a multi-layer artificial neural network to identify admissible parameters. A direct comparison of SNPE and SNL is carried out in [10]. In terms of accuracy, SNL outperforms SNPE in the test cases. However, each approach has its own strengths and weaknesses. Learning the likelihood is often easier and more robust compared to learning the posterior.

[2]Source code available at https://github.com/gpapamak/snl

Furthermore, a model of the likelihood can be reused with different priors. On the other hand, SNPE returns a parametric model of the posterior directly, whereas SNL requires additional inference procedures to compute the posterior, which introduces further computational cost and approximation error. There is perhaps no definite answer to which method is preferable, as the best approach depends on the problem and application at hand.

5 Outlook

This essay provides an overview of likelihood-free inference for parameter identification in mechanistic models and its recent advancements. A central goal of the essay is to familiarize with the world of Bayesian inference and its approximation methods, as they are key subjects and methods for the upcoming master thesis.

Biological neural networks are complex nonlinear dynamical systems, and hence do nonlinear dynamical models play a crucial role in (computational) neuroscience as explanatory tools [14, p. 236]. One such model, the Brunel network model [15], is fit as a problem to apply the (approximate) Bayesian inference for parameter identification discussed in this essay. In particular, both the SNPE and SNL method seem useful for making contact between models and experiments. The Brunel Network model exhibits a high diversity of spiking network dynamics depending on the value of only three synaptic weight parameters. Skaar et al. were able to accurately estimate all network parameters in [16], which may serve as a useful comparison.

The methods presented in this essay are vast topics and the surface has barely been scratched. Below are suggestions for further details and topics, in no particular order, to delve into:

- **ABC: Sampling and rejection methods.** In this essay, two types of ABC methods were discussed: the vanilla *rejection ABC*, and the more sophisticated variant *Markov chain Monte Carlo ABC*. There are, however, more advanced refinements that can be explored further, such as partial least squares dimension reductions for summaries and post-processing regression adjustments. An excellent point of departure for material on improvements is the video lecture by Nott [11].
- **ABC: General pitfalls and remedies.** ABC has been successfully applied to a wide range of problems with an complex or absent associated likelihood. There are, however, several pitfalls to be aware of. The approach is simulation intensive, requires tuning of the tolerance threshold, discrepancy function and weighting function, and suffers from a curse of dimensionality of the summary statistic [5, p. 322]. Therefore, refinements to the methods that may remedy some of the pitfalls should be explored. A point of departure could be [5] and its suggested further reading.
- **Probability as a measure of uncertainty.** One of the most important features of Bayesian inference, or its approximation methods, is that they allow for uncertainty quantification of predictions. One could argue that it is essential in data analysis to not only provide a good model but also an uncertainty estimate of the conclusions, so having this ability might prove beneficial. Points of departure could be [7, e.g. p. 11, 32–34] and the research by Tennøe [17].

- **Choice of (sufficient) summary statistics.** Both the accuracy and computational efficiency of ABC depends on the choice of summary statistics. The guiding principles for constructing effective summary statistics should be investigated. Points of departure could be [7, e.g. ch. 2 and 4] and the paper by Jiang et al. [18].
- **The Brunel Network Model.** Some details of the Brunel network model is provided above, but a more extensive study is needed. In particular, the LFP generated by the network via the hybrid scheme discussed in [16] must be inspected. For more general information on dynamical system models, [14, p. 188, 236–237, 247] can be consulted.
- **Deep learning.** The key idea is to learn the probability density over parameter space. The use of deep learning for this purpose is motivated by two reasons. Neural networks have demonstrated excellent performance in a variety of machine learning problems and deep learning is actively supported by software frameworks such as TensorFlow and PyTorch. Furthermore, in the Bayesian framework, parameters $\boldsymbol{\theta}$ are regarded as random variables. Density estimation based on neural networks scale well with the number of random variables and can incorporate domain knowledge in their design [9]. Specific network configurations to investigate could be the CNN presented by Skaar et al. in [16] and the MDN used by SNPE in [3].

References

[1] Pedro J. Gonçalves et al. "Training deep neural density estimators to identify mechanistic models of neural dynamics". In: *bioRxiv* (2019). DOI: 10.1101/838383. eprint: https://www.biorxiv.org/content/early/2019/11/12/838383.full.pdf. URL: https://www.biorxiv.org/content/early/2019/11/12/838383 (cit. on pp. 1, 8, 9).

[2] George Papamakarios, David C, and Iain Murray. "Sequential Neural Likelihood: Fast Likelihood-free Inference with Autoregressive Flows". English. In: *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics (AISTATS) 2019*. Vol. 89. PMLR, Apr. 2019, pp. 837–848 (cit. on pp. 1, 2, 4, 5, 8, 9).

[3] Jan-Matthis Lueckmann et al. "Flexible statistical inference for mechanistic models of neural dynamics". In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 1289–1299. URL: http://papers.nips.cc/paper/6728-flexible-statistical-inference-for-mechanistic-models-of-neural-dynamics.pdf (cit. on pp. 1, 4, 8, 9, 11).

[4] Mikael Sunnåker et al. "Approximate Bayesian Computation". eng. In: 9.1 (2013), e1002803. ISSN: 1553-734X (cit. on pp. 1, 3–6).

[5] *Handbook of Approximate Bayesian Computation*. eng. 1st ed. CRC Press, 2018. ISBN: 9781439881507 (cit. on pp. 1, 4, 5, 7, 10).

[6] Jay Devore and Kenneth Berk. *Modern Mathematical Statistics with Applications*. eng. Springer Texts in Statistics. New York, NY: Springer New York, 2011. ISBN: 978-1-4614-0390-6 (cit. on pp. 2, 3).

[7] *Bayesian data analysis*. eng. Boca Raton, Fl, 2014 (cit. on pp. 3, 4, 10, 11).

[8] D. S. Sivia and J. Skilling. *Data Analysis - A Bayesian Tutorial*. 2nd. Oxford Science Publications. Oxford University Press, 2006 (cit. on p. 3).

[9] George Papamakarios. *Neural Density Estimation and Likelihood-free Inference*. 2019. arXiv: 1910.13233 [stat.ML] (cit. on pp. 4–7, 11).

[10] Conor Durkan, George Papamakarios, and Iain Murray. *Sequential Neural Methods for Likelihood-free Inference*. 2018. arXiv: 1811.08723 [stat.ML] (cit. on pp. 5, 9).

[11] David Nott. *Approximate Bayesian Computation (ABC)*. https://www.youtube.com/watch?v=77yECJ8_dyk. Video Lecture Accessed 04/15/20. 2018 (cit. on pp. 7, 10).

[12] George Papamakarios and Iain Murray. *Fast ϵ-free Inference of Simulation Models with Bayesian Conditional Density Estimation*. 2016. arXiv: 1605.06376 [stat.ML] (cit. on p. 8).

[13] David S. Greenberg, Marcel Nonnenmacher, and Jakob H. Macke. *Automatic Posterior Transformation for Likelihood-Free Inference*. 2019. arXiv: 1905.07488 [cs.LG] (cit. on pp. 8, 9).

[14] Daniel Durstewitz. *Advanced Data Analysis in Neuroscience: Integrating Statistical and Computational Models*. eng. Bernstein Series in Computational Neuroscience. Cham: Springer International Publishing, 2017. ISBN: 978-3-319-59974-8 (cit. on pp. 10, 11).

[15] Nicolas Brunel. "Dynamics of Sparsely Connected Networks of Excitatory and Inhibitory Spiking Neurons". eng. In: *Journal of Computational Neuroscience* 8.3 (2000), pp. 183–208. ISSN: 0929-5313 (cit. on p. 10).

[16] Jan-Eirik W. Skaar et al. "Estimation of neural network model parameters from local field potentials (LFPs)". In: *bioRxiv* (2019). DOI: 10.1101/564765. eprint: https://www.biorxiv.org/content/early/2019/03/01/564765.full.pdf. URL: https://www.biorxiv.org/content/early/2019/03/01/564765 (cit. on pp. 10, 11).

[17] Simen Tennøe. *Uncertainty quantification in neuroscience*. eng. 2019. URL: http://urn.nb.no/URN:NBN:no-71541 (cit. on p. 10).

[18] Wing Wong et al. “Learning Summary Statistic for Approximate Bayesian Computation via Deep Neural Network”. In: *Statistica Sinica* (2018). ISSN: 1017-0405. DOI: 10.5705/ss.202015.0340. URL: http://dx.doi.org/10.5705/ss.202015.0340 (cit. on p. 11).