Bayesian Modelling of Epidemic Processes

D. Hutchinson

April 18, 2021

Dedication

Accompanying Resources

Abstract

Contents

1 Introduction **1**

2 Bayesian Modelling **2**

2.1 Bayes Factor 2

3 Approximate Bayesian Computation **3**

3.1 Motivation and Background 3

3.2 ABC Methods 4

3.2.1 ABC-Rejection Sampling 9

3.2.2 ABC-Importance Sampling 11

3.2.3 ABC-MCMC 13

3.2.4 ABC-SMC 17

3.2.5 Comparison 21

3.3 ABC for Model Choice 21

3.4 Regression Adjustment in ABC 23

4 Summary Statistic Selection **24**

4.1 Motivation 24

4.2 Properties of Summary Statistics 25

4.3 Methods for Summary Statistic Selection 34

4.3.1 Approximate Sufficient Subset 34

4.3.2 Minimising Entropy 36

4.3.3 Two-Step Minimum Entropy 38

4.3.4 Semi-Automatic ABC 40

4.3.5 Non-Linear Projection 43

4.3.6 Toy Example 44

4.4 Model Selection 44

5 ABC and Epidemic Events **45**

5.1 SIR Model 45

6 Conclusion **46**

6.1 Future Areas of Research 46

Bibliography **i**

1 Introduction

What is a model? A (simple) mathematical formulation of a process which incorporates parameters of interest and likely some stochastic processes. Models need to be computational tractable (i.e. fairly simple)

"All models are wrong, some are useful".

What to use models for? check intuition, explanation & prediction.

What is "posterior estimation"?

The problem - Posterior estimation when likelihood is intractable. "Likelihood-free" estimation. (Classical example of determining most recent common ancestor of two DNA strands. Likelihood is intractable due to number of branches growing factorially. ([Burr and Skurikhin, 2013])

Motivation

What is bayesian inference

Bayes Rule? Describe each component & why is likelihood intractable?

Why now? More, better data. Greater computational power.

What can posterior be used for?

Generative models?

Motivating Examples

DNA mutation ([Marjoram and Tavaré, 2006])

History

Traditional parameter estimation methods - "Maximum Likelihood".

Neutrality testing - (Hypothesis testing), compare results against a null hypothesis for a parameter value.

Successful Applications of these Methods

2 Bayesian Modelling

Bayes' Rule

Bayes rule allows for explanation of relationships in data (rather than just inferences)

Define Bayesian inference

vs. Frequentist modelling

Stochastic vs deterministic models

Consistency

In general, we never know if our calculated posterior is actually close to the true posterior.

Typically only have access to one set of observations (e.g. time-series of covid cases)

Prior encodes assumptions/knowledge so reduces variance but can introduce bias

Theorem 2.1 (Bayes' Rule)
Consider two random variables X and Y. Bayes' Rule provides a formulation for the conditional distribution of A given B.

$$\mathbb{P}(Y|X) = \frac{\mathbb{P}(X|Y)\mathbb{P}(Y)}{\mathbb{P}(X)}$$

where each component is known as

- $\mathbb{P}(Y|X)$*, the Posterior of* Y *given variable* X*.*
- $\mathbb{P}(X|Y)$*, the Likelihood of* Y *given fixed event* X*.*
- $\mathbb{P}(X)$*, the prior distribution of* X*.*
- $\mathbb{P}(Y)$*, the evidence for fixed event* Y*.*

Proof. Bayes' rule is follows from the definition of conditional distributions and joint distributions

$$\begin{aligned}\mathbb{P}(Y|X) &= \frac{\mathbb{P}(X,Y)}{\mathbb{P}(Y)} \\ &= \frac{\mathbb{P}(Y|X)\mathbb{P}(X)}{\mathbb{P}(Y)}\end{aligned}$$

□

Even if the theorised model is not very close to the theorised model (e.g. may only be accurate for a subset of the response space etc.) these inferences are still useful as long as the limitations of the theorised model are well understood.

Priors

2.1 Bayes Factor

3 Approximate Bayesian Computation

In this section I motivate and provide the mathematical background for Approximate Bayesian Computation (ABC) methods *Section 3.1*; Present the general approach of ABC methods *Section 3.2* and discuss four flavours of ABC algorithm *Section 3.2.1-3.2.4*; Provide a comparison of these four methods *Section 3.2.5*; and, I close this section by exploring how ABC methods can be used for model choice *Section 3.3* and how regression adjustment can be used to improve the results of ABC methods *Section 3.4*.

3.1 Motivation and Background

Consider a model X with parameters θ. The centre-point of Bayesian inference is the posterior distribution $\mathbb{P}(\theta|X)$ for the parameters θ given observations X. Using Bayes rule we have the following formulation for this posterior .

$$\mathbb{P}(\theta|X) = \frac{\mathbb{P}(X|\theta)\mathbb{P}(\theta)}{\mathbb{P}(X)}$$

For Bayesian inference we are only concerned with the relative weight the posterior assigns to each parameter value θ, so we can discard the evidence $\mathbb{P}(X)$ as it is just a normalising constant with respect to θ. Meaning we can simplify the expression for the posterior as being proportional to the product of the likelihood $\mathbb{P}(X|\theta)$ and the prior $\mathbb{P}(\theta)$.

$$\mathbb{P}(\theta|X) \propto \mathbb{P}(X|\theta)\mathbb{P}(\theta)$$

As the prior is defined by the user, the only remaining task is to deduce an expression for the likelihood. However, for most real-world processes an explicit expression of the likelihood is computationally intractable due to the complex nature of the systems which govern them and their high degrees of freedom. Moreover, there are often so many parameters that it is intractable to specify all of them and thus we generally theorise a simpler model $\hat{X}$ and seek to calibrate this model to the true model by fitting its parameters. This motivates the need for likelihood-free inference methods such as Approximate Bayesian Computation.

Suppose you have a sequence of n of observations $x_{obs} := (x_{obs,1}, \ldots, x_{obs,n})$ from our model X where each observation may be multi-dimensional, $x_{obs,i} \in \mathbb{R}^p$ for $p \in \mathbb{N}$. Let $K_\varepsilon(\cdot)$ denote a kernel density function with bandwidth $\varepsilon > 0$ and $\|\cdot\|$ denote a distance measure between observations of model X. I discuss kernel density functions and distance measures in *Section 3.2*. Note that as the bandwidth tends to zero the value of the kernel density function for the distance between two points $K_\varepsilon(\|x - x_{obs}\|)$ tends to the Dirac delta function $\delta_{x_{obs}}(x)$. This result is trivially from the definition of a kernel density function.

$$\lim_{\varepsilon \to 0} K_\varepsilon(\|x - x_{obs}\|) = \delta_{x_{obs}}(x) := \begin{cases} 1 & \text{if } x_{obs} = x \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

This result can be used to restate the likelihood function in terms of a kernel density function and distance measure.

$$\begin{aligned} \mathbb{P}(x_{obs}|\theta) &= \int \delta_{x_{obs}}(x)\mathbb{P}(x|\theta)dx \\ &= \lim_{\varepsilon \to 0} \int K_\varepsilon(\|x - x_{obs}\|)\mathbb{P}(x|\theta)dx \end{aligned}$$

Consider the following definition π_{ABC} and note that it tends to, within a normalising constant, of the true posterior.

$$\begin{aligned} & & \pi_{ABC}(\theta|x_{obs}) &:= \int K_\varepsilon(\|x - x_{obs}\|)\mathbb{P}(x|\theta)\pi_0(\theta)dx \\ \implies & \lim_{\varepsilon\to 0}\pi_{ABC}(\theta|x_{obs}) &= \lim_{\varepsilon\to 0}\int K_\varepsilon(\|x - x_{obs}\|)\mathbb{P}(x|\theta)\pi_0(\theta)dx \\ & &= \int \delta_{x_{obs}}(x)\mathbb{P}(x|\theta)dx \cdot \pi_0(\theta) \\ & &= \mathbb{P}(x_{obs}|\theta)\pi_0(\theta) \\ & &\propto \mathbb{P}(\theta|x_{obs}) \end{aligned}$$

This shows that π_{ABC} is an approximation of the true posterior, with it being a good approximation when ε is small.

Typically, due to the observations x being of high dimension, a summary statistic $s(\cdot)$ is applied to them first and then the quantities $s := s(x), s_{obs} := (x_{obs})$ are used in place of x, x_{obs}. The analysis of the above derivation is unchanged when using summary statistics as long as the summary statistics are sufficient, if the summary statistics are not sufficient then π_{ABC} can only ever be an approximation of the true posterior regardless of the bandwidth used. *Section 4* is dedicated to the topic of how to approach choosing summary statistics, with sufficiency being discussed in *Section 4.2*.

$$\pi_{ABC}(\theta|s_{obs}) := \int K_\varepsilon(\|s - s_{obs}\|)\mathbb{P}(s|\theta)\pi_0(\theta)ds$$

This formulation for the ABC approximation of the posterior is the one found in the standard ABC framework (e.g. [Sisson *et al.*, 2018; Beaumont, 2019]).

The utility of being able to use $\pi_{ABC}(\theta|s_{obs})$ to approximate the true posterior is apparent when you consider the implied joint distribution of parameters and summary statistics $\pi_{ABC}(\theta, s|s_{obs})$

$$\begin{aligned} \pi_{ABC}(\theta|s_{obs}) &= \int \pi_{ABC}(\theta, s|s_{obs})ds \\ \text{where}\quad \pi_{ABC}(\theta, s|s_{obs}) &:= K_\varepsilon(\|s - s_{obs}\|)\mathbb{P}(s|\theta)\pi_0(\theta) \end{aligned}$$

We can define Monte Carlo algorithms which target sampling from this joint distribution without needing to specify the likelihood $\mathbb{P}(s|\theta)$. These samples become samples from the posterior by simply ignoring the summary statistic values s which are sampled.

3.2 ABC Methods

Approximate Bayesian Computation (ABC) methods are a family computational methods which can be used to approximate posteriors for the parameters of models where the likelihood is intractable. This is achieved by simulating from the likelihood, rather than having to evaluate it explicitly.

The first algorithm to use the concept which would later be known as ABC was presented in [Tavaré *et al.*, 1997], although this algorithm does not include the use of summary statistics nor use distance measures and kernel density functions to determine whether to accept a simulation or not. The algorithm presented in [Pritchard *et al.*, 1999] is much more recognisable as ABC and consider by many as the first true ABC algorithm. This algorithm would later be generalised to becomes the rejection sampling approach to ABC. Both of these papers were studies of population genetics, a field in which ABC is still popular used.

The key feature that simulation based methods exploit is that we only know the response values x_{obs} from the true model, but for each simulation we know the variables values and the response values $(\tilde{\theta}, \tilde{x})$. Thus we can inspect the parameter values for accepted simulations and draw inferences about the parameter values of the true model.

The central concept for all ABC methods is that the likelihood function can be approximated by comparing simulated values to values from a true model. ABC methods require a set of observations from the true model; a theorised model for which parameters can be set and observations generated; and a set of priors for the parameters of the theorised model. ABC methods then perform many simulations of the theorised model and, by comparing the summary statistic values of the simulated observations to those of the true observations, inferences are made about which parameter values are most likely to be closest to the true values. **Algorithm 3.1** outlines this basic flow which ABC methods follow. The general idea being that the parameter sets which make the theorised model generate observations which are closest to true observations are more likely to be the true parameter values.

Algorithm 3.1 (Generic Approximate Bayesian Computation)
Require: *Observed values x_{obs}; Summary statistics $s(\cdot)$; Theorised model $f(X|\cdot)$; Acceptance Kernel $K_\varepsilon(\cdot)$; Distance Measure $\|\cdot\|$.*

1. *Calculate summary statistic values $s_{obs} = s(x_{obs})$.*
2. *Until stopping condition reached:*
   (a) *Sample a set of parameters $\tilde{\theta}$.*
   (b) *Run the theorised model with sampled parameter $\tilde{x} = f\tilde{\theta}(X|\tilde{\theta})$.*
   (c) *Calculate summary statistic values $\tilde{s} = s(\tilde{x})$.*
   (d) *Accepted parameters $\tilde{\theta}$ with probability $K_\varepsilon(\|\tilde{s} - s_{obs}\|)$.*
3. *Return all accepted parameter sets $\hat{\Theta}$.*

Algorithm 3.1 demonstrates the simplicity of the underlying algorithm for ABC methods. Most ABC methods are straightforward to implement as they follow this basic structure and then change how certain parts of performed in practice (Typically how new samples are drawn and how the acceptance criteria are defined). This allows for a high level of modularity which has motivated innovations in ABC methods.

There are two sources of approximation in the standard ABC algorithm: Use of summary statistics; and, using a bandwidth on the acceptance criteria. The first can be removed by using sufficient summary statistics (See *Section 4.2*). The second is eliminated if the bandwidth is set to zero $\varepsilon = 0$ but in general this leads to the algorithms becoming intractable.

The ideal ABC methods are those which run efficiently and perform well with small bandwidths ε. Efficient methods are important as this means more simulations can be processed in a given time-period, making convergence of the estimated posterior more likely. A method being able to handle smaller bandwidths means the posterior it produces will be a better approximation of the true posterior (See Eq. 1). All ABC methods will run with any value of the bandwidth, however those that use an informed search method for generating samples will require fewer simulations to achieve good results (e.g. ABC-SMC).

Monte Carlo methods are a family of algorithms which use repeated random simulations to evaluate a model. These form the basis of how ABC methods approach exploring the parameter space. Monte Carlo methods are a class of methods which seek to generate samples from a space in a way which mimics sampling from the true model. They do this by running many, many

simulations and use some degree of randomness to determine how each simulation is generated and which are accepted.

Here is an overview of classes of Monte Carlo methods which are commonly used in ABC methods:

- *Rejection-Sampling methods* calculate a probability p that a given set of simulated values came from the true model. A value $u \sim U[0,1]$ is sampled from standard uniform distribution and if the sampled value u is less than the acceptance-probability p then the simulation is accepted as a sample. This procedure is run on a large number of simulations with each simulation being generated and assessed independently.

- *Importance-Sampling methods* extend rejection-sampling by, instead of only accepting a subset of simulated values, all simulations are accepted but each is assigned a weight which indicates the perceived probability that that simulation could be generated by the true model. Typically this weight is the same as the acceptance probability p calculated in rejection-sampling.

- *Markov Chain Monte Carlo (MCMC) methods* extend rejection-sampling by, instead of generating each simulation independently, the parameters of the last accepted simulation are slightly perturbed and then used to generate a new simulation. This creates a search process rather than random simulation due to the dependency between consecutive samples.

- *Sequential Monte Carlo (SMC) methods*[1] extend importance-sampling by repeatedly resampling from the set of samples, with the weights of each parameter determining the probability it is sampled, and each iteration tightening the acceptance criteria. This means the estimated posterior will become more refined each time and hopefully converge on the true posterior.

The use of Monte Carlo methods means that ABC methods are inherently computationally inefficient due to the need to perform many, many simulations to . Further, Monte Carlo methods introduce a high degree of randomness into ABC methods which further motivates the need to perform lots of simulations as the strong law of large number is required to obtain consistent results. This limitation is mitigated due to the simplicity of most ABC algorithms meaning they are capable of process millions of simulations an hour on modern computers.

The set of accepted parameter sets $\hat{\Theta}$ returned by ABC can be used for Bayesian inference. Estimating properties of the distributions, such as mean, mode and quantiles, is straightforward. Producing a discretised estimate of the posterior for each parameter can be achieved by calculating a histogram of the accepted values for each parameter, again straightforward. Kernel density functions can be used to produce a continuous estimates of the posteriors (See [Zambom and Dias, 2012]).

Remark 3.1 (Posterior Mean is Minimum Mean-Square Error Estimator)
Let θ denote the quantity we wish to estimate, A denote an arbitrary estimator of θ and suppose we have observed x_{obs} from model X. Then

$$\begin{aligned} MSE_\theta(A) &= \mathbb{E}\left[(\theta - A)^2 | X = x_{obs}\right] \\ &= \mathbb{E}\left[\theta^2 - 2A\theta + A^2 | X = x_{obs}\right] \\ &= \mathbb{E}\left[\theta^2 | X = x_{obs}\right] - 2A\mathbb{E}\left[\theta | X = x_{obs}\right] + A^2 \\ \implies \frac{\partial}{\partial A} MSE_\theta(A) &= -2\mathbb{E}[\theta | X = x_{obs}] + 2a \\ \implies a &= \mathbb{E}[\theta | X = x_{obs}] \end{aligned}$$

[1] Also known as Particle-Filter methods.

This shows that mean-square error is minimised when the posterior mean of θ given x_{obs} is used as an estimator.

ABC methods are commonly used to calibrate models or to compare models. Typically calibration is done by setting parameter values to the estimated posterior mean as the posterior mean minimises mean-square error (see **Remark 3.1**). ABC methods are used for model comparison as they can directly estimate Bayes factor, I discuss model comparison further in *Section 3.3*.

The key advantage of ABC methods, over other approaches to Bayesian inference, is that it produces a distribution, rather than a point-estimate, for parameter values. This allows for analysis into uncertainty around the parameter values. Additionally, as the strictness of the acceptance criteria is a parameter of ABC methods, ABC methods can fit or compare a large range of theorised models by loosening the acceptance criteria. Being able to use simpler models has the advantage of reducing issues which occur due to curse-of-dimensionality.

A limitation of using ABC methods is the large number of hyper-parameters they have (Distance measures, summary statistics, bandwidths, perturbation kernels, etc.) and that the choices the user makes for how these parameters are set can drastically affect the algorithms performance. It is trivial to realise that if an uninformative distance measure such as $\|x\| = 0 \; \forall \; x$ is used or an acceptance kernel which accepts all simulations is used then the returned set of parameters will resemble the set of priors, and no meaningful inferences can be drawn. Moreover, these hyper-parameters need to be tuned for each model these methods are applied, which is laborious. This has motivated the innovation of adaptable ABC algorithms which automate the process of setting some of these parameters.

As stochastic processes determine whether a simulation is accepted, or not, ABC methods incur information loss. This can mean that promising areas of the parameter space are not explored. This issue is mitigate by running many simulations.

Summary Statistics

See *Section 4*.

Kernel Density Functions

Definition 3.1 (Kernel Density Functions $K_\varepsilon(\cdot)$, Epanechnikov [1969])
Kernel density functions are functions $K : \mathbb{R} \to \mathbb{R}$ with the following properties:

1. *Non-negative*
$$K_\varepsilon(x) \geq 0 \; \forall \; x \in \mathcal{X}$$
where $\mathcal{X}$ is the range of values x can take.

2. *Symmetric*
$$K_\varepsilon(x) = K_\varepsilon(-x) \; \forall \; x \in \mathcal{X}$$

3. *Normalised*
$$\int_{\mathcal{X}} K_\varepsilon(x)dx = 1$$

4. $K_\varepsilon(x) = \frac{1}{\varepsilon} K_1(x/\varepsilon)$.

Name	Formula
Uniform Kernel	$K_\varepsilon(x) = \frac{1}{2\varepsilon}\mathbb{1}\{x \leq \varepsilon\}$
Gaussian Kernel	$K_\varepsilon(x) = \frac{\varepsilon}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}x^2\varepsilon^2\right\}$
Epanechnikov Kernel	$K_\varepsilon(x) = \frac{3}{4}\left(1 - x^2\varepsilon^2\right)\mathbb{1}\left\{\lvert x\rvert \leq \varepsilon\right\}$

Table 1: Common kernel density functions for ABC methods.[2]

Kernel density functions are typically extended to allow for a smoothing parameter $\varepsilon \geq 0$ such that $K_\varepsilon(x) = \frac{1}{\varepsilon}K(x/\varepsilon)$.

The choice of kernel density function does not play a notable role in the asymptotic behaviour of ABC methods, however the bandwidth chosen for them does. A high bandwidth means that the weight of the kernel is spread much more evenly across its support meaning there is less discrimination between values close to the mean and those further away.

It is standard to define kernel density functions such that they have zero mean. Having this property means that that $\max_x K_\varepsilon(x) = K_\varepsilon(0)$, this follows immediately from the kernel being symmetric. This is a useful property in the context of ABC methods as we pass the distance between two points $\|x - x_{obs}\|$ to the kernel density function to determine the probability we accept a simulation and this property means that simulations x closest to the observed values x_{obs} are more likely to be accepted.

In practice, when implementing ABC methods we typically scale up the values returned by the kernel such that $K_\varepsilon(0) = 1$. This is straightforward to do for well-known kernels as it only requires the removal of the normalising term. As the relative weights given to each value are maintained this does not affect the asymptotic behaviour of the algorithms, but will increase the acceptance rate. This also has the desirable effect that every time an exact match is found it will definitely be accepted.

Table 1 provides a table of the most commonly used kernel density functions for ABC, as recommended by [Beaumont, 2019]. The Epanechnikov kernel is asymptotically optimal for kernel density estimation when seeking to minimise mean-square error (See [Epanechnikov, 1969]), although this theoretical result is disputed in [Tsybakov, 2008].

The uniform kernel is popular in ABC as it is equivalent to accepting all simulations whose distance from the true observation is no greater than ε. This creates spherical acceptance regions when using the Euclidean distance or rectangular ones when using Manhattan distance. The Gaussian kernel is more commonly used as it has an infinite support which is useful in certain scenarios. When choosing which kernel to use for ABC methods it is intuitive that it should match the theorised distribution of the noise in the theorised model. This further motivates the popularity of a gaussian kernel as many models assume gaussian noise.

Distance Measures

Distance measures quantify how far apart two multi-dimensional vectors are from each other, with greater values indicating the vectors are further away. A value of zero means that the two vectors are identical under the given measure.

The choice of distance measure is integral to the performance of an ABC method as it determines whether a set of simulated values are deemed to be representative of the true model,

[2] $\mathbb{1}\{A\} := \begin{cases} 1 & \text{if } A \\ 0 & \text{otherwise} \end{cases}$

Name	Formula
Manhattan Distance	$L_1(\mathbf{x}) := \sum_{i=1}^{m} \lvert x_i \rvert$
Euclidean Distance	$L_2(\mathbf{x}) := \sqrt{\sum_{i=1}^{m} x_i^2}$
L_p Norm	$L_p(\cdot) := \left(\sum_{i=1}^{m} x_i^p \right)^{1/p}$
L_∞ Norm	$L_\infty(\mathbf{x}) := \max\{x \in \mathbf{x}\}$

Table 2: Common distance measures for ABC methods.

or not, by quantifying how similar these values are to true observations. **Table 2** provides a list of popular distance measures for ABC methods. The Euclidean distance is most commonly as minimising Euclidean distance is clearly related to minimise SSE of a model.

It is important to note that the value passed to distances measures in ABC methods is the difference of two sets of summary statistics. Well chosen summary statistics will extract meaningful information and perform a certain level of pre-processing of this data such as standardisation and weighting different dimensions. This means we do not need to consider these problems during selection of distance measure.

An issue which arises when specifying distance measures is the "Curse of Dimensionality" [3]. This is the phenomena that as the dimensionality of vectors being compared increases, it becomes harder to distinguish between different pairs. This is an issue to ABC methods as the success of the approach relies on being able to accurately identify which simulations are closest to observed values. Using summary statistics which introduce a high level of dimensionality reduction will help.

Different demonstrations of this phemonema is required for different distance measures, but for Euclidean distance it is generally demonstrated by comparing of the volume of a hyper-sphere with radius r and the volume of a hyper-cube with side length $2r$. The volume of the hyper-cube quickly dwarfs that of the hyper-sphere as the number of dimensions are increased. The "Curse of Dimensionality" is a big short coming of the Euclidean distance as it was only every conceived for real-world spaces (i.e. two or three dimensions). Which distance measure is best ultimately depends on the data being used. Using the L_p norm has shown promise but adds the additional problem of what value of p is optimal. [Schnitzer *et al.*, 2014] present an approach to choosing an optimal p which assesses the "hub-ness" of a dataset.

There is a wealth of literature on the "Curse of Dimensionality" in the machine learning space, particularly for nearest-neighbour problems which are very relevant to the problem being addressed by distance measures in ABC (See [Beyer *et al.*, 1999; Hinneburg *et al.*, 2000; Radovanovic; *et al.*, 2010])

3.2.1 ABC-Rejection Sampling

ABC-Rejection Sampling is a generalisation of the sampling algorithms presented in [Tavaré *et al.*, 1997; Pritchard *et al.*, 1999]. The general idea is to keep simulating from the theorised model until a predefined number of simulations M have been accepted by the acceptance kernel. Each simulation involves sampling a set of parameters $\tilde{\theta}$ from the predefined priors $\pi_0(\theta)$; initialising the theorised model $f(X|\theta)$ with the sampled parameters; Observing values $\tilde{x}$

[3] Term first coined in [Bellman, 1961] in reference to how many algorithms may work well when applied to low-dimensions, but are intractabled when for higher-dimensions.

from the initialised model and then comparing these observations to observations from the true model using summary statistics s, a distance measure $\|\cdot\|$ and a acceptance kernel $K_\varepsilon(\cdot)$. This approach is stated formally in **Algorithm 3.2**.

ABC-rejection sampling is most suitable in problems where it is believed that the posterior is not very different from the defined priors, as this algorithm has very limited search capabilities. These are typically problems which have already been studied heavily an a general understanding of the priors is known.

Algorithm 3.2 (ABC-Rejection Sampling "Fixed Sample Size", Beaumont *et al.* [2002])

require: *Observed values* x_{obs}*; Summary statistics* $s(\cdot)$*; Theorised model* $f(X|\cdot)$*; Prior Distributions* $\pi_0(\theta)$*; Acceptance Kernel* $K_\varepsilon(\cdot)$*; Distance Measure* $\|\cdot\|$*; Target Number* M*.*

$s_{obs} \leftarrow s(x_{obs})$.
$\tilde{\Theta} \leftarrow \{\}$.
$t \leftarrow 0$.
while $t < M$ **do**
  $\tilde{\theta}_t \leftarrow$ *sample* $\pi_0(\theta)$.
  $\tilde{x} \leftarrow f(X|\tilde{\theta}_t)$.
  $\tilde{s} \leftarrow s(\tilde{x})$.
  with probability $K_\varepsilon(\|s_{obs} - \tilde{s}\|)$
    $\hat{\theta}^{(t)} \leftarrow \tilde{\theta}$.
    Add $\hat{\theta}^t$ *to* $\hat{\theta}$.
    $t \leftarrow t + 1$
  otherwise *Pass;*
return $\tilde{\Theta} = \left\{\theta^{(1)}, \ldots, \theta^{(M)}\right\}$

The approach in **Algorithm 3.2** is intuitive and simple to implement. A limitation of this simplicity is that each simulation is completely independent and no "learning" is incorporated from information about which parameter sets have previously been accepted, or rejected. However, this independence does mean that it is straight forward to implement **Algorithm 3.2** in a parellisable fashion, allowing for more simulations to be analysed in a given time-period.

A practical difficulty in using **Algorithm 3.2** is in setting the bandwidth of the acceptance kernel as the bandwidth dictates the acceptance rate λ. A lower acceptance rate means more meaningful inferences can be drawn from the results as the average distance between accepted simulations and the true observations will be lower, and thus the accepted simulations will be more informative. However, there is no clear relationship between bandwidth and the acceptance rate of the algorithm so run-times are near-impossible to predict to any meaningful degree of accuracy.

Algorithm 3.3 (ABC-Rejection Sampling "Best Samples")

require: *Observed values x_{obs}; Summary statistics $s(\cdot)$; Theorised model $f(X|\cdot)$; Prior Distributions $\pi_0(\theta)$ Distance Measure $\|\cdot\|$; Number of Simulations M; Simulations to Accepted N.*

$s_{obs} \leftarrow s(x_{obs})$.
$\tilde{\Theta} \leftarrow \{\}$.
$t \leftarrow 0$.
for $i = 0, \ldots, M$ **do**
  $\tilde{\theta}^{(i)} \leftarrow$ *sample* $\pi_0(\theta)$.
  $\tilde{x}^{(i)} \leftarrow f(X|\tilde{\theta}^{(i)})$.
  $\tilde{s}^{(i)} \leftarrow s\left(\tilde{x}^{(i)}\right)$.
  $d^{(i)} \leftarrow \|s(\tilde{x}^{(i)}) - s_{obs}\|$.
  Add $(d^{(i)}, \tilde{\theta}^{(i)})$ *to* $\tilde{\Theta}$.
return N *elements with smallest distance values* $d^{(i)}$.

The problem of setting a bandwidth (and acceptance kernel) can be avoid completely by instead running a fixed number M of simulations and then accepting a predefined number N of these which are closest to the observed values, effectively defining the acceptance rate. Additionally, this removes the risk of the algorithm running indefinitely. This approach is outlined in **Algorithm 3.3**. This algorithm runs in linear time with-respect-to the number of simulations $O(M)$, as the set of simulated values $\tilde{\Theta}$ is unsorted with-respect-to the distance values $d^{(i)}$ and thus finding the N^{th} order-statistic takes linear time.

The space requirements for **Algorithm 3.3** grow linearly with-respect-to the number of simulations being run $O(M)$. This creates a pratical limit of the number of simulations which can be run. The space requirements can be reduced to $O(N)$ (grow linearly wrt the number of accepted simulations) by, instead of storing all simulations in the set $\tilde{\Theta}$, we instead only store the N closest. This requires keeping the set $\tilde{\Theta}$ ordered and thus increases the time complexity of the algorithm to $O(M \log_2 N)$.

There is no requirement for the number of simulations, nor the number to accept, to be defined for running the algorithm. Rather the algorithm can be allowed to run and assess simulations until a time limit is reached. Then either a predefined proportion of the simulations can be accepted, or the distribution of distances can be inspected to choose an acceptance rate. The disadvantage of this approach is that it has high space requirements due to the need to store the distance and parameters for all simulations until the very end of the algorithm. This creates a cap on how long this version of the algorithm could be run, but this can be mitigated by dropping the very worst simulations (among other approaches).

The set of accepted simulations returned by ABC-Rejection Sampling techniques places an equal weighting on each simulation. A natural extension is to place greater weight on parameters which produce values which are closer to those produced by the true model. [Beaumont *et al.*, 2002] propose a technique of weighted local-linear regression to adjust parameter values where the weights are determined by the distance value associated with the parameter set.

Rejection sampling techniques can be used to estimate the probability of a given set of results under different models $f(X|M)$. This is useful in model choice as it immediately leads to estimations of Bayes' factor. I discuss this further in *Section 3.3*.

3.2.2 ABC-Importance Sampling

Importance sampling methods use a tractable distribution to sample from an intractable distribution. Importance sampling is an exact method as, given enough iterations, it will always converge on target distribution in an unbiased fashion. The theory behind importance

sampling is laid out in **Remark 3.2**.

Remark 3.2 (Importance Sampling)
Let X be a model $X \sim h(X;\theta)$ with parameters θ. Consider two distributions for parameter values $f(\theta), g(\theta)$ where f is a target distribution, from which sampling is intractable, and g is a distribution we can sample from. Then the expected value of X under distribution f is the same as the expected value of $X \cdot w(\theta)$ under distribution g, where $w(\theta) := \frac{f(\theta)}{g(\theta)}$ is a weight measure.

$$\begin{aligned}
\mathbb{E}_f[X] &= \int h(X;\theta) f(\theta) d\theta \\
&= \int h(X;\theta) f(\theta) \frac{g(\theta)}{g(\theta)} d\theta \\
&= \int h(X;\theta) g(\theta) \frac{f(\theta)}{g(\theta)} d\theta \\
&= \int h(X;\theta) g(\theta) w(\theta) d\theta \text{ where } w(\theta) := \frac{f(\theta)}{g(\theta)} \\
&= \mathbb{E}_g[h(X;\theta) w(\theta)]
\end{aligned}$$

This means we can estimate the expected value of the model under f by weighting samples from g using $w(\theta)$. This is the likelihood ratio of an observation coming from the two models.

In ABC context the target distribution is the approximate joint distribution posterior $f = \pi_{ABC}(\theta, s|s_{obs})$ and the distribution we can sample from is the joint distribution of summary statistics and parameters under the importance distribution $g(s,\theta) = p(s|\theta)g(\theta)$. The importance weighting $\tilde{w}$ for each simulation in ABC is derived below.

$$\begin{aligned}
\frac{\pi_{ABC}(\theta, s|s_{obs})}{g(\theta, s)} &\propto \frac{K_\varepsilon\left(\|s - s_{obs}\|\right) p(s|\theta)\pi_0(\theta)}{p(s|\theta)g(\theta)} \\
&= \frac{K_\varepsilon\left(\|s - s_{obs}\|\right) \pi_0(\theta)}{g(\theta)} \\
&=: \tilde{w}
\end{aligned}$$

ABC-Importance Sampling should be used in cases where we have a good idea of what the distribution for the posterior will be so that the prior π_0 and importance distribution g are informative.

An importance sampling approach to ABC is an extension of the rejection sampling approach which replaces calculating acceptance probabilities with calculating importance weightings to each simulation. All simulations are accepted and their importance weight is used to weight them during Bayesian inference. An acceptance kernel K_ε and distance measure $\|\cdot\|$ still need to be specified as they are required to calculate the importance weights. This approach is given in **Algorithm 3.4**.

Algorithm 3.4 (ABC-Importance Sampling)

require: *Observed values x_{obs}; Summary statistics $s(\cdot)$; Theorised model $f(X|\cdot)$; Prior Distributions $\pi_0(\theta)$ Distance Measure $\|\cdot\|$; Number of Simulations M; Importance Kernel $g(\cdot)$.*

1 $s_{obs} \leftarrow s(x_{obs})$.
2 $\tilde{\Theta} \leftarrow \{\}$.
3 **for** $i = 0, \ldots, M$ **do**
4 $\quad \tilde{\theta}^{(i)} \leftarrow$ *sample* $g(\theta)$.
5 $\quad \tilde{x}^{(i)} \leftarrow f(X|\tilde{\theta}^{(i)})$.
6 $\quad \tilde{s}^{(i)} \leftarrow s\left(\tilde{x}^{(i)}\right)$.
7 $\quad \tilde{w}^{(i)} \leftarrow \frac{\pi_0\left(\theta^{(i)}\right)}{g\left(\theta^{(i)}\right)} K_\varepsilon\left(\left\|s^{(i)} - s_{obs}\right\|\right)$.
8 $\quad$ *Add* $\tilde{\theta}^{(i)}$ *to* $\tilde{\Theta}$ *with weight* $\tilde{w}^{(i)}$.
9 **return** $\tilde{\Theta} := \left\{\left(\tilde{\theta}^{(1)}, \tilde{w}^{(1)}\right), \ldots, \left(\tilde{\theta}^{(M)}, \tilde{w}^{(M)}\right)\right\}$

Similar to ABC-Rejection Sampling, **Algorithm 3.4** is straightforward to implement in a parallelisable fashion due to the independence of each simulation. However, it is much less space efficient than the ABC-Rejection Sampling approaches as it requires the storage of every simulation. This is mitigated by an approach presented by [Fearnhead and Prangle, 2011] which combines the rejection and importance sampling approaches to ABC. I discuss this approach more further down.

The approach to ABC-Importance Sampling given in **Algorithm 3.4** requires the specification of priors $\pi_0(\theta)$ and an importance distribution $g(\theta)$. If these distributions are the same, or proportional to each other, then $\frac{\pi_0(\theta)}{g(\theta)} \approx 1 \ \forall \ \theta$ meaning the acceptance probability $K_\varepsilon(\|s - s_{obs}\|)$ is the only factor weighting each simulation.

An issue with all sampling approaches which weight their results is that it is possible for a small subset of accepted samples to dominate the weight space. This can lead to results becoming unstable. This can naturally be tackled by increasing the number of simulations, this is inefficient and does not inform us as to when a sufficient number of simulations have been made. The Effective Sample Size (ESS) is a useful metric in these cases as it quantifies how many equally weighted samples our set is equivalent to. The stopping condition of the algorithm should be updated such that the algorithm terminates once the effective sample size of the accepted set of parameters has reached some threshold N

$$ESS := \frac{\sum_{i=0}^{M} w^{(i)}}{\sum_{i=0}^{M} \left(w^{(i)}\right)^2}$$

where $(w^{(0)}, \ldots, w^{(M)})$ is the weights assigned to each simulation.

[Fearnhead and Prangle, 2011] propose an algorithm which combines ABC-Rejection Sampling and ABC-Importance Sampling by, rather than accepting every simulation (Line 8), each simulation is accepted with probability $K_\varepsilon(\|s - s_{obs}\|)$ and is assigned weight $\tilde{w} = \pi_0(\theta)/g(\theta)$. This reduces the accepted set of simulations to only those that produce reasonably similar responses, compared to the true model. This improves the effective sample size of the set of accepted parameters as fewer simulations are given very small weights, and is more space efficient that **Algorithm 3.4** as it does not require every simulation to be stored.

3.2.3 ABC-MCMC

Definition 3.2 (Markov Chain)
A Markov Chain is a Stochastic Process $\{X_t\}_t$ with the Markov Property. This means that the current state of the process solely depends on its state in the time-period immediately before.

$$\mathbb{P}(X_{t+1}|X_t, \ldots, X_1) = \mathbb{P}(X_{t+1}|X_t)$$

The transitions a Markov Chain can make can be summarised in a square matrix P_t, known as the "transition matrix", where $[P_t]_{ij} = \mathbb{P}(X_{t+1} = j|X_t = i)$. The transition matrix can be time invariant.

A Markov chain is said to be "irreducible" if it is possible to go from any state to any other state, in some finite period of time.

$$\mathbb{P}(X_{t+n} = x|X_t = y) > 0 \ \forall \ x, y$$

The Stationary Distribution of a Markov Chain is a probability distribution π which is invariant under a time-invariant transition matrix P.

$$\pi = \pi P$$

The stationary distribution represents the asymptotic proportion of time the chain spends in each state. The stationary distribution is unique if the Markov chain is irreducible.

Markov chains are sequences of events where the probability of which event occurs next only depends on the current event. When targeting a probability distribution the transition matrix for a Markov chain will be stationary, this means it will have a stationary distribution which can be approximated.

Markov Chain Monte Carlo (MCMC) methods are sampling methods which exploit Markov chains in order to have a more informed search procedure through the parameter space. The Markov chain is used to determine which set of parameters to simulate with next, with the next choice being dependent upon the most recently accepted set of parameters. An acceptance step, similar to ABC-Rejection sampling, is then used to evaluate the simulated response values against the true model values and thus whether to accept the new set of parameters. The distribution of accepted parameter sets is an approximation of the stationary distribution of the Markov chain, and thus of the target distribution. These algorithms are ideally run until the distribution of accepted samples satisfies some convergence criteria, although in practice it is more practical to stop the algorithm once the chain has reached a given length.

A popular class of MCMC algorithms are Metropolis-Hastings algorithms [Metropolis *et al.*, 1953; Hastings, 1970] which seeks to produce a Markov chain whose stationary distribution is unique and thus converges on the target distribution (The parameter posterior in the case of ABC methods). This approaches requires the specification of a perturbation kernel $K^*(\theta)$ which generates a new set of parameters by slightly perturbing a given set of parameters. The perturbance kernel needs to implemented in such a way that the probability of it generating a given set of parameters θ', given the input θ, is calculatable.

$$\mathbb{P}\left(K^*(\theta) = \theta'\right)$$

The simplest perturbation kernels apply additive gaussian noise to the input, the variance on the noise is a hyper-parameter which would require tuning. More complex perturbation kernels consider the correlation between parameters and then step correlated parameters in the same/opposite direction. Fisher information can be incorporated into perturbation kernels in order to determine which parameters have a greater effect and thus should be explored more.

[Filippi *et al.*, 2012] explore selecting perturbation kernels for ABC-SMC but many of the themes are relevant to ABC-MCMC too.

$$K^*(\theta) = \theta + \mathcal{N}(0, \sigma_0^2) \text{ for some } \sigma_0^2 \geq 0$$

As each sample is drawn using the previously accepted sample, there is dependence between samples leaving MCMC methods open to auto-correlation issues. Auto-correlation is a measure of correlation between the current value of a sample and its previous values. Auto-correlation can be reduced by increasing the size of steps the perturbation kernel is expected to produce but this will have adverse affects on the acceptance rate. Auto-correlation can be particularly high if the chain becomes stuck in a region where there is very concentrated probability mass as it will struggle to escape. The problem with auto-correlation is that most analysis assumes that parameters are independent, but auto-correlation can contradict this assumption.

A limitation of MCMC methods is that they are only able to search one region of the sample space at any given time and they struggle to move between disconnected areas of high density. In the context of Bayesian inference, this causes an issue when wishing to model multi-modal distributions as MCMC will typically only be able to find one of the modes. The solution to this is to run multiple chains at once and then to merge their results. This does, however, require greater computational resources and typically means that each chain is made shorter to compensate.

MCMC methods have limited scope for being parallelised as each iteration depends on the previous iteration. If multiple chains are being run, then these can be parallelised.

[Marjoram *et al.*, 2003] presents the first ABC method to have an MCMC approach, using the popular Metropolis-Hastings. **Algorithm 3.5** presents their algorithm. This approaches has two main stages: An initial burn-in (Lines 4-7) where random sets of parameters are evaluated until one is found which is accepted by the standard acceptance criteria used in ABC-Rejection Sampling; and, the MCMC step (Lines 9-19) which starts at the first accepted parameter set $\tilde{\theta}_0$ and proceeds to generate new parameter sets θ^* by perturbing the last accepted parameter set. These new parameters sets are then used to generate simulations, and are accepted with probability $\min\left\{1, \frac{K_\varepsilon(\|s^* - s_{obs}\|)\pi(\theta^*)\mathbb{P}\left(K^*(\tilde{\theta}^{(t-1)}) = \theta^*\right)}{K_\varepsilon(\|s^{(t-1)} - s_{obs}\|)\pi(\theta^{(t-1)})\mathbb{P}\left(K^*(\theta^*) = \tilde{\theta}^{(t-1)}\right)}\right\}$[4].

Algorithm 3.5 (ABC-MCMC, Marjoram *et al.* [2003])

[4]This probability is known as the "Metropolis Acceptance Ratio" and was derived so that the stationary distribution of the Markov chain will converge on the target distribution.

require: *Observed values* x_{obs}*; Summary statistics* $s(\cdot)$*; Theorised model* $f(X|\cdot)$*; Prior Distributions* $\pi_0(\theta)$ *Distance Measure* $\|\cdot\|$*; Chain length* M*; Acceptance Kernel* $K_\varepsilon(\cdot)$*; Perturbation Kernel* $K^*(\cdot)$*.*

1 $s_{obs} \leftarrow s(x_{obs})$.
2 $\tilde{\Theta} \leftarrow \{\}$.
3 *# Burn-In Step*
4 **while** $K_\varepsilon\left(\left\|\tilde{s}^{(0)} - s_{obs}\right\|\right)$ *is not accepted* **do**
5 $\quad \tilde{\theta}_0 \leftarrow$ *sample* $\pi_0(\theta)$.
6 $\quad \tilde{x}^{(0)} \leftarrow f(X|\tilde{\theta}^{(0)})$.
7 $\quad \tilde{s}^{(0)} \leftarrow s\left(\tilde{x}^{(0)}\right)$
8 *# MCMC Step*
9 **for** $t = 1, \ldots, M$ **do**
10 $\quad \theta^* \leftarrow K^*\left(\tilde{\theta}^{(t-1)}\right)$.
11 $\quad x^* \leftarrow f\left(X|\theta^*\right)$.
12 $\quad s^* \leftarrow s\left(x^*\right)$.
13 $\quad$ **with probability** $\min\left\{1, \frac{K_\varepsilon(\|s^* - s_{obs}\|)\pi(\theta^*)\mathbb{P}\left(K^*(\tilde{\theta}^{(t-1)}) = \theta^*\right)}{K_\varepsilon(\|s^{(t-1)} - s_{obs}\|)\pi(\theta^{(t-1)})\mathbb{P}\left(K^*(\theta^*) = \tilde{\theta}^{(t-1)}\right)}\right\}$
14 $\quad\quad \tilde{\theta}^{(t)} \leftarrow \theta^*$.
15 $\quad\quad s^{(t)} \leftarrow s^*$.
16 $\quad$ **otherwise**
17 $\quad\quad \tilde{\theta}^{(t)} \leftarrow \tilde{\theta}^{(t-1)}$.
18 $\quad\quad s^{(t)} \leftarrow s^{(t-1)}$.
19 $\quad$ *Add* $\tilde{\theta}^{(t)}$ *to* $\tilde{\Theta}$.
20 **return** $\tilde{\Theta} := \left\{\tilde{\theta}^{(1)}, \ldots, \tilde{\theta}^{(M)}\right\}$

The approach **Algorithm 3.5** choose to stop the MCMC step after a set number of iterations. This is not a good choice as it does not consider whether the stationary distribution of the Markov chain has converged. There are a few empirical methods which can be implemented to assess convergence. [Gelman and Rubin, 1992] propose running multiple chains, with different starting locations, and assessing the ratio of intra-chain to inter-chain variance for each parameter. When this ratio is close to one then convergence has been achieved. This method is not always practical to use due to its requirement for multiple chains and in practice we often choose to run the algorithm until some time-limit is reached.

The burn-in period (Lines 4-7) is equivalent to running ABC-Rejection Sampling until the first set of parameters is accepted. Thus it is liable to running for an indeterminant amount of time (potentially indefinitely) and the solution is the same as for ABC-Rejection Sampling: run a fixed number of simulations and choose the best one. This approach can be extended to automate the setting of the bandwidth used in the MCMC step (Lines 9-19), which can otherwise be a difficult task during tuning. The burn-in step is a crucial part of the algorithm as if the Markov chain does not start in an area of high posterior density then the rest of the algorithm will perform very badly. It is often necessary to run multiple burn in simultaneously in order to chose a more informed starting location.

The algorithm can be made more adaptable by having it actively update the perturbation kernel to maintain a target acceptance rate. In the case of an additive gaussian noise kernel, increasing the variance should lead to a decrease in acceptance rate as it is more likely that large steps will be taken. The acceptance rate can also be managed by adaptively setting the bandwidth on the acceptance kernel used in calculating the Metropolis acceptance ratio.

The acceptance rate of an MCMC methods controls the rate of convergence, with both too high and too low values leading to slow convergence. An ideal acceptance rate will how a good level of mixing, so that the parameter space is explored efficiently. It was shown in [Gelman *et al.*, 1997] that the asympotitcally optimal acceptance rate for a Metropolis-Hasting sampler is 0.234, as the number of dimensions tends to infinity, when the target distribution is Gaussian. This result does rely on each dimension being independent identically distribution gaussian distributions, which is not always reasonable. Study into some more general in-homogeneous target distributions have also shown 0.234 to be the asymptotically optimal acceptance rate (See [Roberts and Rosenthal, 2001]) but a general result has yet to be found. This research does motivate the use of adaptive MCMC methods which target an acceptance rate of 23.4%.

Due to their more informed search procedure, ABC-MCMC significantly outperforms ABC-Rejection and Importance Sampling in cases where the prior and posterior are very different. This makes ABC-MCMC a better choice in cases were informative priors are not known. However, the ABC-MCMC approach perform very poorly with mixtures models, which are becoming increasingly popular, due to the "Label Switching Problem" [Jasra *et al.*, 2005] which occurs when two, or more, parameters are nonidentifiable when assigned the same priors[5] and thus the posteriors produced for them will be a combination of all of their true posteriors. [Jasra *et al.*, 2005] explore the "Label Switching Problem".

3.2.4 ABC-SMC

Sequential Monte Carlo (SMC) methods[6] approximate a probability distribution by collecting an initial sample which creates a rough approximation of the distribution; and then iteratively refining this approximation by resampling under ever tighter acceptance criteria (Referred to as improving the "Resolution" of the approximation). The acceptance criteria are tightened by defining a set of bandwidths $\{\varepsilon_0, \ldots, \varepsilon_T\}$ such that $\varepsilon_0 \geq \cdots \geq \varepsilon_T$ and iterating through this set to determine the bandwidth used in each resampling step. A extension of this approach is to incorporate importance sampling such that the resampling step also involves reweighting accepted samples. This extension is known as Population Monte Carlo (PMC).

The main advantage of SMC methods is that they iteratively make their acceptance criteria stricter. This is ideal for problems where it is hard to predict a good set of acceptance criteria beforehand. There is still an issue of having to define a set of bandwidths $\{\varepsilon_0, \ldots, \varepsilon_T\}$ to be used, however I discuss how this can be mitigated for ABC-SMC towards the end of this subsection.

SMC methods are susceptible to "Loss of Opportunity". This phenomenon occurs when part of the parameter space is not included in one of the sample sets, as this means that part of the parameter space can never be sampled from in the future. This mainly occurs to regions of the parameter space where little probability mass is placed, but can occur to denser areas if the sample size is too small. This issue can never be eliminated, except for very simple distributions, do the practical limits on the sample size but can be mitigated by increasing the sample size.

[Sisson *et al.*, 2007] presents the first SMC approach to ABC, but this approaches produces a biased approximation of the posterior, mainly due to it underestimating the tails of the distributions caused by how they originally proposed to evaluate the likelihood ratio. [Beaumont *et al.*, 2009] presents an SMC approach to ABC which incorporates importance sampling and an optimised adaptive strategy. This is the version of ABC-SMC I discuss in this section.

[5] In a gaussian mixtures model with two mixtures. The parameters associated with each mean can be swapped without affecting the fit of the model. This means that under identical priors it is impossible to separate these two parameters.

[6] Originally coined Particle Filters in [Del Moral, 1997].

Algorithm 3.6 (ABC-SMC, Beaumont *et al.* [2009])

require: *Observed values x_{obs}; Summary statistics $s(\cdot)$; Theorised model $f(X|\cdot)$; Prior Distributions $\pi_0(\theta)$ Distance Measure $\|\cdot\|$; Acceptance Kernel $K_\varepsilon(\cdot)$; Set of Bandwidths $\{\varepsilon_0, \ldots, \varepsilon_T\}$; Number of Iterations T; Sample Size N.*

$s_{obs} \leftarrow s(x_{obs})$.

 # Initial Sample Step

$\tilde{\Theta}_0 \leftarrow \{\}$.

$i \leftarrow 0$

while $i < N$ **do**

  $\tilde{\theta}_0^{(i)} \leftarrow$ *sample* $\pi_0(\theta)$.

  $\tilde{x}_0^{(i)} \leftarrow f(X|\tilde{\theta}_t^{(i)})$.

  $\tilde{s}_0^{(i)} \leftarrow s\left(\tilde{x}_t^{(i)}\right)$.

  with probability $K_{\varepsilon_0}\left(\left\|\tilde{s}_0^{(i)} - s_{obs}\right\|\right)$

    $w_0^{(i)} \leftarrow \dfrac{1}{N}$.

    Add $\tilde{\theta}_0^{(i)}$ *to* $\tilde{\Theta}_0$ *with weight* $\tilde{w}_0^{(i)}$.

    $i \leftarrow i + 1$

  otherwise *Pass;*

 # Resampling Step

for $T = 1, \ldots, T$ **do**

  $\sigma^2_{t-1} \leftarrow$*Sample variance of each parameter dimension in* $\tilde{\Theta}_{t-1}$.

  $K^* \leftarrow$ *Normal*$(\theta, 2 \cdot \sigma^2_{t-1})$.

  $\tilde{\Theta}_t \leftarrow \{\}$.

  $i \leftarrow 0$

  while $i < N$ **do**

    $\tilde{\theta}_t^{(i)} \leftarrow$ *sample* $\tilde{\Theta}_{t-1}$.

    $\theta^* \leftarrow K_t^*\left(\tilde{\theta}_t^{(i)}\right)$.

    $\tilde{x}_t^{(i)} \leftarrow f(X|\theta^*)$.

    $\tilde{s}_t^{(i)} \leftarrow s\left(\tilde{x}_t^{(i)}\right)$.

    with probability $K_{\varepsilon_t}\left(\left\|\tilde{s}_t^{(i)} - s_{obs}\right\|\right)$

      $\tilde{\theta}_t^{(i)} \leftarrow \theta^*$.

      $\tilde{w}_t^{(i)} \leftarrow \dfrac{\pi_0\left(\tilde{\theta}_t^{(i)}\right)}{\sum_{j=1}^N w_{t-1}^{(j)} \mathbb{P}\left(K_t^*\left(\tilde{\theta}_{t-1}^{(j)}\right) = \tilde{\theta}_t^{(i)}\right)}$.

      Add $\tilde{\theta}_t^{(i)}$ *to* $\tilde{\Theta}_t$ *with weight* $\tilde{w}_t^{(i)}$.

      $i \leftarrow i + 1$.

    otherwise *Pass;*

  # Normalise Weights

  for $i = 1, \ldots, N$ **do**

    $w_t^{(i)} \leftarrow \dfrac{\tilde{w}_t^{(i)}}{\sum_{i=1}^N \tilde{w}_t^{(i)}}$.

    Update weight of $\tilde{\theta}_t^{(i)}$ *in* $\tilde{\Theta}_t$ *to be* $w_t^{(i)}$.

return $\tilde{\Theta}_T := \left\{\left(\tilde{\theta}_T^{(1)}, w_T^{(1)}\right), \ldots, \left(\tilde{\theta}_T^{(N)}, w_T^{(N)}\right)\right\}$

Algorithm 3.6 is the algorithm presented in [Beaumont *et al.*, 2009]. This algorithm has two phases: First, (Lines 3-13) generating an initial sample of parameters $\tilde{\Theta}_0$ of size N using standard ABC-Rejection Sampling methods. Each sample is assigned the same importance weight $1/N$; Second, the *Resampling Step* (Lines 15-34). This involves resampling from the previously set of accepted parameter samples $\tilde{\Theta}_{t-1}$ with the probability of sampling each parameter equal to its importance weight. Each sample $\tilde{\theta}$ is perturbed using a perturbation kernel $K^*(\cdot)$ to generate a new set of parameters θ^*. The new parameter set θ^* is used to simulate a set of summary statistic values $\tilde{s}$ and a rejection sampling step is used to accepted the new parameter set with probability $K_{\varepsilon_t}(\|\tilde{s} - s_{obs}\|)$. Note that the acceptance criteria are tightened each iteration. Each accepted parameter set is assigned an importance weight $\tilde{w}$. The importance weights are normalised after each resampling phase so that they sum to one and thus represent a probability distribution, which is important for sampling from this set.

The importance weight $\tilde{w}$ assigned in Line 27 is the prior probability for the accepted parameter set divided by the probability of that parameter set under the posterior $\hat{\pi}_t$ generated by the previous step. This is just the standard importance weighting of the likelihood ratio. Note that each resampling step is aiming to produce a more refined version of the posterior distribution generated by the previous step, and thus the previous distribution is the target distribution and the prior is the originally proposed distribution.

$$\tilde{w}_t^{(i)} := \frac{\pi_0\left(\tilde{\theta}_t^{(i)}\right)}{\hat{\pi}_t\left(\theta_t^{(i)}\right)} \text{ where } \hat{\pi}_t\left(\theta_t^{(i)}\right) = \sum_{j=1}^{N} w_{t-1}^{(j)} \mathbb{P}\left(K_t^*\left(\tilde{\theta}_{t-1}^{(j)}\right) = \tilde{\theta}_t^{(i)}\right)$$

The adaptive feature of **Algorithm 3.6** is the setting of the perturbance kernel K^*. The perturbance kernel used in **Algorithm 3.6** is a component-wise random walk kernel which perturbs each component of the parameter set independently by adding additive gaussian noise to them. The variance of this gaussian noise is equal to twice the sample variance of the accepted samples from the previous phase.

$$\left[\sigma_{t-1}^2\right]_i = \frac{1}{N-1} \sum_{j=1}^{N} \left(\left[\tilde{\theta}_{t-1}^{(j)}\right]_i - \left[\bar{\theta}_{t-1}\right]_i\right)^2$$

where $\bar{\theta}_{t-1}$ is the sample mean of the previous set of accepted samples.

Using a component-wise random walk kernel is ideal for an adaptive algorithm as it is straightforward to implement and is computationally efficient as simple closed-form expressions for the probabilities required to calculate the importance weight for each accepted parameter set.

The variance is set to twice the sample variance of the previously accepted set as this minimises the Kullback-Leibler divergence between the target distribution (two independent parameter samples) and proposed distribution (generating a set of parameters by perturbing another) for the component-wise random walk kernel being used. Minimising Kullback-Leibler divergence means the two distributions are increasingly similar. See [Filippi *et al.*, 2012] for discussion of other optimal perturbation kernels for ABC-SMC.

The calculation of the importance weight for each accepted parameter, during resampling, requires summing over all the parameter sets from the previously accepted sample set. This means the resample stage takes $O(N^2)$ time and thus the overall run time of the algorithm is $O(TN^2)$ where T is the number of resampling iterations and N is the sample size. In practice the runtime of the algorithm will be dominated by assessing and generating samples, as the majority will be rejected, rather than by calculating the weight for each accepted set.

Each resampling step is dependent on the previous step as it requires the previous set of

accepted samples $\tilde{\Theta}_{t-1}$ in order to generate samples. This means that this part of the algorithm cannot be parallelised. However, the simulations within each resampling step can be parallelised. As well as the initial sample generation step, as discussed in *Section 3.2.1.*

This approach requires the specification of a set of bandwidths $\{\varepsilon_0, \ldots, \varepsilon_T\}$. This can be difficult to do in an informed way, and would rather be avoided. Firstly, it is important to note that it is not strictly necessary for the algorithm to use the whole set and rather the algorithm can be stopped after it has reached a certain level of convergence (or number of simulations). Further, there is no need to define a full set of bandwidths at the start of the algorithm, instead an initial bandwidth ε_0 can be defined and then future bandwidths are set adaptively such that a target percentage $\Delta\%$ of previously sample would be accepted. Implementing this is straightforward for most common acceptance kernels, if a uniform acceptance kernel is being used it simply requires setting the next bandwidth to be the Δ percentile distance among the previously accepted parameter sets.

The need to set the initial bandwidth can be removed too by simply accepting all simulations into the initial sample set $\tilde{\Theta}_0$, however this would make the algorithm significantly more inefficient as the initial sample with simple resemble the prior. Further, unless the sample size is very large there will be a high level of "Loss of Opportunity". A better approach would be to use the "Best Samples" variation of ABC-Rejection Sampling (**Algorithm 3.3**).

Incorporating this adaptive approach to bandwidth selection removes the need to define a set of bandwidths $\{\varepsilon_0, \ldots, \varepsilon_T\}$ or the number of iterations T; and replaces them with defining an acceptance rate and number of simulations for the "Initial Sample Step", a target acceptance rate between resampling iterations and a maximum number of simulations. These are significantly easier hyperparameters to define as their affects are much more apparent and predictable.

3.2.5 Comparison

Which algorithm to use in different scenarios - complexity of model, amount of data available.

I compare each algorithm's performance using a toy SIR model example. See *Section 5.1* for a formal definition and discussion of the SIR model.

3.3 ABC for Model Choice

Definition 3.3 (Bayes Factor, Kass and Raftery [1995])
Consider two models M_1, M_2 and some observed data x_{obs}. The Bayes Factor $B_{1,2}$ for data x_{obs} coming from model M_1 rather than model M_2 is the ratio of the likelihood ratio of x_{obs} coming from M_1 rather than M_2.

$$B_{1,2} := \frac{\mathbb{P}(x_{obs}|M_1)}{\mathbb{P}(x_{obs}|M_2)}$$

[Jeffreys, 1961] gives a qualitative assessment of Bayes Factor: "1 to 3 is barely worth a mention, 3 to 10 is substantial, 10 to 30 is strong, 30 to 100 is very strong and over a 100 is decisive evidence in favour of model M_1. Values below 1 take the inverted interpretation in favour of model M_2."

Bayes Factor is a metric used to determine which of two models is more likely to have generated some observed data. Bayes Factor can be restated in terms of posteriors, using Bayes rule, as follows.

$$B_{1,2}(x_{obs}) := \frac{\mathbb{P}(x_{obs}|M_1)}{\mathbb{P}(x_{obs}|M_2)} = \frac{\frac{\mathbb{P}(x_{obs})\mathbb{P}(M_1|x_{obs})}{\mathbb{P}(M_1)}}{\frac{\mathbb{P}(x_{obs})\mathbb{P}(M_2|x_{obs})}{\mathbb{P}(M_2)}} = \frac{\mathbb{P}(M_2)\mathbb{P}(M_1|x_{obs})}{\mathbb{P}(M_1)\mathbb{P}(M_2|x_{obs})} \tag{2}$$

where $\mathbb{P}(M_i)$ is the prior weight given to model M_i. It is generally reasonable to assume equal prior likelihood for each model. Under this assumption Bayes factor is the same as the posterior ratio which is readily estimatable from ABC methods as the ratio of probabilities that the models generate x_{obs}.

$$B_{1,2}(x_{obs}) = \frac{\mathbb{P}(M_1|x_{obs})}{\mathbb{P}(M_2|x_{obs})}$$

Algorithm 3.7 (ABC Model Selection "Rejection Sampling", Grelaud *et al.* [2009])

require: *Observed values x_{obs}; Summary statistics $s(\cdot)$; Priors for Models $\pi_M(M)$; Theorised models $M_1(X|\theta_{M_1}), M_2(X|\theta_{M_2})$; Parameter Priors for each model $\pi_{M_1}(\theta_{M_1}), \pi_{M_2}(\theta_{M_2})$; Acceptance Bandwidth ε; Distance Measure $\|\cdot\|$; Target Number M.*

$s_{obs} \leftarrow s(x_{obs})$.
$\mathcal{M} \leftarrow \{\}$.
$t \leftarrow 1$.
while $t \leq M$ **do**
  $m_t \leftarrow$ *sample* $\pi_M(M)$.
  $\tilde{\theta}_t \leftarrow$ *sample* $\pi_{m_t}(\theta)$.
  $\tilde{x} \leftarrow f(X|\tilde{\theta}_t)$.
  $\tilde{s} \leftarrow s(\tilde{x})$.
  if $\|s_{obs} - \tilde{s}\| \leq \varepsilon$ **then**
    $\hat{\theta}^{(t)} \leftarrow \tilde{\theta}$.
    Add m_t to $\mathcal{M}$.
    $t \leftarrow t + 1$
  otherwise *Pass;*
return $\mathcal{M} = \{m_1, \ldots, m_M\}$

[Grelaud *et al.*, 2009] present an algorithm which uses an alteration of the ABC-Rejection Sampling algorithm, using a uniform kernel, to estimate Bayes Factor. Their approach is outlined in **Algorithm 3.7**. This approach defines a meta-model $M = (M_1, M_2)$ which is a mixtures model which uses model M_1 or M_2 according to some distribution π_M. The distribution π_M indicates our prior belief of the likelihood of each model. The algorithm then proceeds as a standard ABC-Rejection sampling algorithm except during the parameter sampling step it also samples which model to use (this defines which set of parameter priors to use too). Each time a simulation is accepted, the model which generated it is recorded in the set $\mathcal{M}$. The returned set $\mathcal{M}$ provides the ratio of the number of times simulations from each model were accepted which estimates Bayes Factor.

$$\hat{B}_{1,2} = \frac{\sum_{i=1}^{N} \mathbb{1}\{m_i = M_1\}}{\sum_{i=1}^{N} \mathbb{1}\{m_i = M_2\}}$$

The results of **Algorithm 3.7** are sensitive to how informative the priors are for each model and thus can be used to compare different prior sets.

This approach is based on the ABC-Rejection Sampling algorithm and thus does not gain any of the advantages of the ABC-MCMC or ABC-SMC algorithms, namely being effective when the prior and posterior are significantly different. [Toni *et al.*, 2009] present a model selection algorithm which uses ABC-SMC but requires the use of a meta-model which incorporates the models being tested, as in [Grelaud *et al.*, 2009]. [Didelot *et al.*, 2011] present an approach which estimates the evidence for each model independently, using ABC-SMC.

3.4 Regression Adjustment in ABC

Beaumont et al - Local Linear Regressions (LOCL)

Blum and Francois' - Nonlinear Conditional heteroscedastic regressions (NCH). (Uses neural networks)

4 Summary Statistic Selection

In this section I motivate the research into summary statistic selection *Section 4.1* and discuss features to consider when selecting summary statistics *Section 4.2*. I then describe five methods for summary statistic selection methods: three which use hand-crafted summary statistics *Sections 4.3.1-4.3.3*; and two which automatically generate summary statistics *Sections 4.3.4-4.3.5*. These approaches are covered in the chronological order in which they were original proposed. To close the section I use a toy example of an SIR model to compare these methods *Section 4.3.6*.

4.1 Motivation

The study of summary statistics has relevance beyond ABC methods, largely due to the recent "Big-Data Revolution" which has seen the rate at which data can be collected and stored significantly outpace improvements in computational power. This has motivated research into effective methods to reduce the size of datasets so that more computationally intensive algorithms can be used to analyse the data.

A summary statistic s is a statistic which reduces the dimensionality of some sampled data, in a deterministic fashion, whilst retaining as much information about the sampled data as possible. Reducing the dimensionality of data is desirable as it reduces the computational requirements to analyse the data. Ideally, a summary statistic would compress the sampled data without any information loss (A property known as "sufficiency"). However, low-dimension sufficient summary statistics are rare in practice and we often have to trade-off information retention against dimensionality reduction.

$$s : \mathbb{R}^m \to \mathbb{R}^p \text{ with } m > p$$

In most cases each dimension of the output of a summary statistic is the result of an independent calculation. As such, it is often conceptually easier to consider each dimension as an independent summary statistics when selecting summary statistics. This idea of each dimension of independence also makes it conceptually easy to combine summary statistics by appending the result of one statistic onto the end of the other, as new dimensions. As long as the sum of the dimensions of the outputs from the summary statistics in the set is less than that of the sampled data, then using a set of summary statistics still produces effective dimensionality reduction.

$$m > \sum_{i=1}^{k} p_i \text{ where } s_i : \mathbb{R}^m \to \mathbb{R}^{p_i}$$

The success of ABC methods depends mainly on three user choices: choice of summary statistic; choice of distance measure; and choice of acceptance kernel. Of these, summary statistic choice is arguably the most important as the other two mainly affect the rate at which the algorithm converges on the posterior mean. Whereas, choosing summary statistics which are uninformative can lead to the parameter posteriors returned by the algorithm being drastically different from the true parameter posteriors. This is trivial to realise if you consider a scenario where $s(x) = c$, for some constant $c \in \mathbb{R}$, is used as the sole summary statistic as this would result in all (or none) of the simulations being accepted as thus the returned posterior will be the same as the supplied prior.

In practice, the quality of the posteriors returned from an ABC method is limited by the amount of computational time which is dedicated to running the algorithm. For some problems,

such as ..... , it is reasonable to dedicate the majority of your computing time on summary statistic selection, rather than on model fitting, as it is clear that even the simplest ABC methods (e.g. ABC-Rejection Sampling) will be sufficient to fit the model, given a good choice of summary statistics.

Traditional Thinking

Traditionally, summary statistics for ABC methods are chosen manually using expert, domain-specific knowledge. Utilising this expert knowledge is desirable as these statistics will incentivise exploring regions of the parameter space which have been scientifically shown to be relevant to the given problem and thus more likely to contain the true parameter values (Similarly, these statistics will disincentivise exploring regions which have been shown to not be of interest).

However, relying on expert knowledge to choose summary statistics limits the scenarios where ABC methods can be applied to only those where there has already been significant research. And, leads to statistics being chosen due to their prevalence in a field rather than their suitability to computational methods. Moreover, the use of hand-crafted summary statistics means that any limitations in current understanding of a field will be encoded into the model fitting process, possibly leading to misspecification.

When using a set of summary statistics, expert knowledge is generally not sufficient to determine how best to weight each summary statistic. Some of the methods I describe below can be used to automate the process of determining these weights by specifying multiple versions of the same summary statistic, with each version having a different weight.

4.2 Properties of Summary Statistics

When evaluating a summary statistic for use in ABC there are main properties, both practical and mathematical, to consider.

Practical Properties

The key reason for using summary statistics is for the computational efficiencies which result from their dimensionality reduction. Reducing the size of a dataset means less operations need to be performed to analyse it, meaning more simulations can be processed in the same time-period. This naturally means summary statistics which result in greater dimensionality reduction are preferable, but similarly means that a summary statistic which is computationally inefficient to calculate is less desirable.

For a model which produces data of dimension $n \times m$ (i.e. n readings, each with m features) most standard summary statistics are calculated in $O(n \cdot m)$ time. However, this is only a theoretical result and in practice there are meaningful differences in the computational requirements of each summary statistics. Calculating the mean and maximum values for each feature takes $O(n \cdot m)$ time in theory but, since calculating the mean relies on arithmetic operations and the maximum on comparison operations, they will take different amounts of time in practice. Statistics which rely on search or sorting operations (most notably order statistics) are variable in the their time complexity for different data sets which will affect the reliability of models which use them. Integer overflow is a possible issue for some summary statistics, although this is often easy to avoid when actively being considered during the implementation of an algorithm. Moreover, for statistics with non-linear computational complexity (e.g. correlation between each pair of features), the size of the dataset being analysed needs to be considered when evaluating summary statistic choice.

ABC-methods rely on distance measures to determine whether a simulation is good, or not. This means that the range and scale of values a summary statistic will likely produce will have an affect on how influential that summary statistic is to the final model fit. In most cases it is reasonable to standardise all statistics to have the same mean and variance, effectively giving the same weighting to each statistic. This can be implemented to occur adaptively within the ABC-method. There may be cases where assigning different weights to different summary statistics makes sense, and produces a better model fit, but these are hard to justify from a theoretical approach. The selection methods I discuss which compare hand-crafted statistics (Sections 4.3.1-4.3.3) can be used to compare possible weightings by including several versions of the same summary statistic, each with a different scaling, in the set of statistics being compared. This will however increase computation time due to the increase size of the set of statistics and may make the results harder to interpret[7].

For real-world modelling problems, the interpretability of summary statistics used in the final model is a key factor in how useful the solution is. Senior stakeholders in a problem will want to use the final model to justify their future decisions, this is much easier to do when the factors the model is considering, and the weights it assigns to them, are readily understandable. Hand-crafted statistics are almost always the most readily understandable statistics, as such generated statistics are rarely used in commercial problems[8]. In cases where it is chosen to use automatically generated statistics; one can develop an intuition for their model by varying the inputs, or removing certain features, and observing how the output varies. This is naturally harder to

Sufficiency

Definition 4.1 (Sufficient Statistic Casella and Berger [2001])
Let $s : \mathbb{R}^m \to \mathbb{R}^n$ be a statistic and X be a model with parameters θ. The statistic s is said to be sufficient for the parameters θ if the conditional distribution of the model X, given the value of the statistic $s(X)$, is independent of the model parameter.

$$\mathbb{P}(X|s(X)) = \mathbb{P}(X|s(X), \theta)$$

Verbosely, a statistic is sufficient for a model parameter(s) if it captures all the information which a sample of the model carries about said parameter(s). This means that knowing the value of a sufficient statistic is as informative as knowing the true model parameters. This is clearly a desirable property as in practice we can always calculate the value of the summary statistic using the sampled data, but cannot know the true parameter values (otherwise we would not be trying to predict them). Sufficient statistics exist for all models as, trivially, the identity function is a sufficient statistic for all models.

It can be intuitively helpful to consider a sufficient statistic as a data reduction method. Moreover, a sufficient summary statistic provides a loss-less compression of sampled data as it reduces the dimensionality of the data but retains all relevant information.

Remark 4.1 (Supersets of Sufficient Statistics)
Let $s_{1:k-1}(\cdot) := \{s_1(\cdot), \ldots, s_{k-1}(\cdot)\}$ be a collection of $k-1$ summary statistics and suppose

[7] Multiple sets of weighted summary statistics will be equivalent due to having the same ratio of weights

[8] The current popularity of using "Neural Networks" in commercial settings does buck this trend. I hope this fad will subside soon in favour of more interpretable alternatives. I believe it is worth noting that the new European Union payment services directive (PSD2) requires that certain models used by financial institutions be "explainable" in order to improve the customer experience and to ensure no one is discriminated against due to their protected characteristics.

that $s_{1:k-1}$ is sufficient for the parameters θ of some model X. Then $s_{1:k-1} \cup \{s_k\}$ is also sufficient for the parameters θ, for all summary statistics s_k.

Proof. Consider a model with parameters θ and let $s_1, \ldots, s_k$ be summary statistics where the set $s_{1:k-1} := \{s_1, \ldots, s_{k-1}\}$ is sufficient for parameter θ. Note that the likelihood of set $s_k := s_{1:k-1} \cup \{s_k\}$ given the model parameters θ can be stated as

$$\mathbb{P}(s_{1:k}(X)|\theta) = \mathbb{P}(s_k(X)|s_{1:k-1}(X), \theta)\mathbb{P}(s_{1:k-1}|\theta)$$

Now consider the following decomposition of the posterior for the model parameters θ given summary statistics $s_{1:k}$

$$\begin{aligned} \mathbb{P}(\theta|s_{1:k}(X)) &= \frac{\mathbb{P}(s_{1:k}(X)|\theta)\mathbb{P}(\theta)}{\mathbb{P}(s_{1:k}(X))} \\ &= \frac{\mathbb{P}(s_k(X)|s_{1:k-1}(X), \theta)\mathbb{P}(s_{1:k-1}|\theta)\mathbb{P}(\theta)}{\mathbb{P}(s_k(X)|s_{1:k-1}(X))\mathbb{P}(s_{1:k-1}(X))} \end{aligned}$$

Since the set $s_{1:k-1}$ is sufficient for θ we have that

$$\mathbb{P}(s_k(X)|s_{1:k-1}(X)) = \mathbb{P}(s_k(X)|s_{1:k-1}(X), \theta)$$

Applying this result to the decomposition above, we deduce that the posterior for the model parameters θ given $s_{1:k}$ or $s_{1:k-1}$ are identical.

$$\begin{aligned} \mathbb{P}(\theta|s_{1:k}(X)) &= \frac{\mathbb{P}(s_k(X)|s_{1:k-1}(X), \theta)\mathbb{P}(s_{1:k-1}|\theta)\mathbb{P}(\theta)}{\mathbb{P}(s_k(X)|s_{1:k-1}(X), \theta)\mathbb{P}(s_{1:k-1}(X))} \\ &= \frac{\mathbb{P}(s_{1:k-1}|\theta)\mathbb{P}(\theta)}{\mathbb{P}(s_{1:k-1}(X))} \\ &= \mathbb{P}(\theta|s_{1:k-1}(X)) \end{aligned}$$

Thus the set $s_{1:k}$ is sufficient for model parameters θ. Due to the arbitrary nature of $s_{1:k-1}$ and s_k, this result holds for all supersets of sufficient summary statistics. □

Remark 4.1 states that if we have a set of summary statistics which are sufficient for a set of parameters, then adding more summary statistics will never increase (or decrease) the amount of relevant information being extracted from the sampled data. This means there is an optimally minimal number of summary statistics required to achieve sufficiency.

I demonstrate in **Example 4.1** that the sample mean is a sufficient summary statistic for a normal distribution with unknown mean.

Example 4.1 (Sufficient Statistic for Normal Distribution with Unknown Mean)
Let $X \sim Normal(\mu, \sigma_0^2)$, with $\mu \in \mathbb{R}$ unknown and $\sigma_0^2 \in \mathbb{R}$ known, and $\mathbf{x}$ be n independent observations of X.

We have that

$$f_{\mathbf{X}}(\mathbf{X}) = \prod_{i=1}^{n} f_X(X_i) = \frac{1}{(2\pi\sigma_0^2)^{n/2}} \exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}(X_i - \mu)^2\right\}$$

Let $s = s(\mathbf{X})$ be an arbitrary statistic of n observations from the model. We will build up

the conditional distribution of $\mathbf{X}$ *given* $s(\mathbf{X})$*, by first considering their joint distribution*

$$\begin{aligned}
f_{\mathbf{X},s(\mathbf{X})}(\mathbf{X},s) &= \frac{1}{(2\pi\sigma_0^2)^{n/2}}\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}(X_i+s-s-\mu)^2\right\} \\
&= \frac{1}{(2\pi\sigma_0^2)^{n/2}}\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}((X_i+s)-(\mu-s))^2\right\} \\
&= \frac{1}{(2\pi\sigma_0^2)^{n/2}}\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}\left((X_i-s)^2+(\mu-s)^2-2(\mu-s)(X_i-s)\right)\right\} \\
&= \frac{1}{(2\pi\sigma_0^2)^{n/2}}\cdot\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}(X_i-s)^2\right\}\cdot\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}(\mu-s)^2\right\} \\
&\quad \cdot\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}-2(\mu-s)(X_i-s)\right\} \\
&= \frac{1}{(2\pi\sigma_0^2)^{n/2}}\cdot\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}(X_i-s)^2\right\}\cdot\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}(\mu-s)^2\right\} \\
&\quad \cdot\exp\left\{\frac{\mu-s}{\sigma_0^2}\left(\sum_{i=1}^{n}(X_i)-ns\right)\right\}
\end{aligned}$$

If we define $s(\mathbf{X})=\dfrac{1}{n}\displaystyle\sum_{i=1}^{n}X_i$*, the sample mean, then the third exponential disappears. Note that* $s(\mathbf{X})\sim$ *Normal*$\left(\mu,\dfrac{1}{n}\sigma_0^2\right)$*.*

Now consider the conditional distribution of $\mathbf{X}$ *given* $s(\mathbf{X})$*.*

$$\begin{aligned}
f_{\mathbf{X}|s(\mathbf{X})}(\mathbf{X}|s) &= \frac{f_{\mathbf{X},s(\mathbf{X})}(\mathbf{X},s)}{f_{s(\mathbf{X})}(s(\mathbf{X}))} \\
&= \frac{\sqrt{\frac{1}{(2\pi\sigma_0^2)^n}}\cdot\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}(X_i-s)^2\right\}\cdot\exp\left\{-\frac{n}{2\sigma_0^2}(\mu-s)^2\right\}}{\sqrt{\frac{n}{2\pi\sigma_0^2}}\cdot\exp\left\{-\frac{n}{2\sigma_0^2}(\mu-s)^2\right\}} \\
&= \sqrt{\frac{1}{n(2\pi\sigma_0^2)^{n-1}}}\cdot\exp\left\{-\frac{1}{2\sigma_0^2}\sum_{i=1}^{n}(X_i-s)^2\right\}
\end{aligned}$$

This shows that the conditional distribution of $\mathbf{X}$ *given* $s(\mathbf{X})$ *is independent of* μ*, the unknown parameter, and thus the sample mean is a sufficient statistic for a normal distribution with unknown mean*

Example 4.1 shows that finding sufficient summary statistics can be a highly manually and did require us to "guess" at the possible formulation of a summary statistic, then verify that it was sufficient. The Fisher-Neyman factorisation criterion (**Theorem 4.1**) [Fisher, 1922; Neyman, 1935], first recognised by Fisher in [Fisher, 1922], specifies a property which all sufficient statistics have. This property is used as the basis of a more formulaic approach to finding sufficient statistics by separating the terms of the conditional probability of a model given the summary statistic value into those which depend on the summary statistic and those which do not.

Theorem 4.1 (Fisher-Neyman Factorisation Criterion Casella and Berger [2001])
Let $X\sim f(\cdot;\theta)$ *be a model with parameters* θ *and* $s(\cdot)$ *be a statistic.*

$s(\cdot)$ *is a sufficient statistic for the model parameters* θ *iff there exist non-negative*

functions $g(\cdot;\theta)$ and $h(\theta)$ where $h(\cdot)$ is independent of the model parameters[9] *and*

$$f(X;\theta) = h(X)g(s(X);\theta)$$

This formulation shows that the distribution of the model X only depends on the parameter θ through the information extracted by the statistic s. A consequence of the sufficiency of s.

Proof. [Roussas, 1998]

$\Longrightarrow$ First, consider the forwards direction of the theorem and suppose s is a sufficient summary statistic. Define functions

$$h(x) = \mathbb{P}(X = x|s(X) = s(x)) \quad \text{and} \quad g(s(x);\theta) = \mathbb{P}(s(X) = s(x);\theta)$$

Note that $h(\cdot)$ is independent of the model parameter θ due to the sufficiency of s. Then

$$\begin{aligned} f_X(x) &= \mathbb{P}(X = x) \\ &= \mathbb{P}(X = x, s(X) = s(x)) \\ &= \mathbb{P}(X = x|s(X) = s(x))\mathbb{P}(s(X) = s(x)) \\ &= h(X)g(s(X)) \end{aligned}$$

$\Longleftarrow$ Now, consider the reverse direction of the theorem and suppose there exists some functions $h(\cdot), g(\cdot;\theta)$, with $h(\cdot)$ independent of model parameter θ, such that

$$f(x;\theta) = h(x)g(s(x);\theta) \text{ for all } x \in \mathcal{X},\ \theta \in \Theta$$

where $\mathcal{X}$ is the support of X and Θ the set of possible parameters.

Then, for an arbitrary $c \in \mathbb{R}$

$$\begin{aligned} \mathbb{P}(X = x|s(X) = c) &= \frac{\mathbb{P}(X = x, s(X) = c)}{\mathbb{P}(s(X) = c)} \\ &= \frac{\mathbb{1}\{s(x) = c\}f(x;\theta)}{\sum_{y\in\mathcal{X};s(y)=c} f(y;\theta)} \\ &= \frac{\mathbb{1}\{s(x) = c\}h(x)g(s(x);\theta)}{\sum_{y\in\mathcal{X};s(y)=c} h(y)g(s(y);\theta)} \\ &= \frac{h(x)g(c;\theta)}{\sum_{y\in\mathcal{X};s(y)=c} h(y)g(c;\theta)} \\ &= \frac{h(x)}{\sum_{y\in\mathcal{X};s(y)=c} h(y)} \end{aligned}$$

This final expression is independent of the model parameter θ.

The result holds in both directions. □

[9] i.e. $h(\cdot)$ only depends on the sampled data

Example 4.2 below demonstrates how the Fisher-Neyman Factorisation Theorem can be used to find a sufficient summary statistic for a Poisson model where the mean λ is unknown

Example 4.2 (Using Fisher-Neyman Factorisation Theorem to find sufficient statistics for a Poisson distribution with unknown mean)
Let $X \sim$ Poisson(λ), with $\lambda \in \mathbb{R}^{>}$ unknown, $\mathbf{x}$ be n independent observations of X and $\bar{x} := \frac{1}{n}\sum_{i=1}^{n} x_i$ be the sample mean of these n observations.

Consider the joint distribution of these n observations

$$\begin{aligned} f_{\mathbf{X}}(\mathbf{x}) &= \prod_{i=1}^{n} f_X(x_i) \\ &= \prod_{i=1}^{n} \frac{\theta^{x_i} e^{-\theta}}{x_i!} \\ &= \frac{1}{\prod_{i=1}^{n} x_i!} \cdot \theta^{\sum_{i=1}^{n} x_i} e^{-n\theta} \\ &= \underbrace{\left\{ \frac{1}{\prod_{i=1}^{n} x_i!} \right\}}_{(1)} \cdot \underbrace{\left\{ \theta^{\sum_{i=1}^{n} x_i} e^{-n\theta} \right\}}_{(2)} \end{aligned}$$

The last step shows how the terms can be collected into: (1), those which are independent of model parameter θ; and, (2), those which are dependent on model parameter θ. We can how derive the conditions of the Fisher-Neyman Factorisation theorem by inspecting the final expression.

It is apparent that we should define the function $h(\mathbf{x})$ as

$$h(\mathbf{x}) = \frac{1}{\prod_{i=1}^{n} x_i!}$$

In order to define the function $g(s(\mathbf{x}); \theta)$ we first need to define the summary statistic $s(\mathbf{x})$. This is straightforward as all the sampled data $\mathbf{x}$ only occurs in a sum in (2), so we define $s(\mathbf{x}) = \sum_{i=1}^{n} x_i$. Meaning we can define $g(\mathbf{x}; \theta)$ as

$$g(\mathbf{x}; \theta) = \theta^{s(\mathbf{x})} e^{-n\theta}$$

With these definitions we fulfil the conditions of the Fisher-Neyman Factorisation theorem, meaning $s(\mathbf{X}) = \sum_{i=1}^{n} X_i$ is a sufficient statistic for the mean for a Poisson distribution.

In most cases sufficient statistics for a parameter are not unique. Moreover, each sufficient statistic does not necessarily produce the same level of compression. Consider a normal distribution with unknown mean, here both the sample sum and identity function are both sufficient statistics, however the sample sum is a much more desirable statistic to use as it provides compression down to a single dimension. This lack of uniqueness motivates the concept of minimal sufficiency.

Definition 4.2 (Minimally Sufficient Statistic, Dodge *et al.* [2006])
Let $s(\cdot)$ be a sufficient statistic for parameter θ of model X. $s(\cdot)$ is minimally sufficient if for any other sufficient statistic $t(\cdot)$ of parameter θ there exists a function f which maps $t(x) \mapsto s(x)$.

$$s(X) = f(t(X))$$

Minimally sufficient statistics have lower (effective) dimensionality than their non-minimal counterparts. This makes minimally sufficient statistics desirable as they produce the greatest level of compression and, in doing so, maximally reduce the computational resources required to analyse the sampled data.

As with identifying sufficient statistics, determining whether, or not, a sufficient statistic is minimally sufficient is not a trivial task. I demonstrate this in **Example 4.3**.

Example 4.3 (Minimally Sufficient Statistic for IID Bernoulli Random Variables)
Let $X_1, \ldots, X_n$ are independent and identically distribution Bernoulli random variables. Note that the identity function $s_1(\mathbf{X}) = \mathbf{X}$ and the sum function $s_2(\mathbf{X}) = \sum_{i=1}^{n} X_i$ are both sufficient statistics.

We can map from s_1 to s_2 as follows

$$s_2(\mathbf{X}) = \sum_{i=1}^{n} [s_2(\mathbf{X})]_i$$

However, there is no function which can map from s_2 to s_1 as it would have to map the value 1 to both $(1, 0, \ldots, 0)$ and $(0, 1, \ldots, 0)$. This proves that the identity function s_1 is not a minimally sufficient statistic, but does not prove that the sum function s_2 is a minimally sufficient statistic as we have not considered all possible sufficient statistics for this distribution.

Theorem 4.2 (Condition for Minimal Sufficiency, Balakrishnan [2019])
Consider a model with parameters θ. Let $\mathbf{x}, \mathbf{y}$ be two samples from this model and $s(\cdot)$ be a statistic.

If $\frac{\mathbb{P}(\mathbf{y};\theta)}{\mathbb{P}(\mathbf{x};\theta)}$ is independent of θ iff $s(\mathbf{x}) = s(\mathbf{y})$, then statistic s is minimally sufficient.

Proof. Let $s(\cdot)$ be a statistic for model X with parameters θ and assume that $\frac{\mathbb{P}(\mathbf{y};\theta)}{\mathbb{P}(\mathbf{x};\theta)}$ is independent of θ iff $s(\mathbf{y}) = s(\mathbf{x})$. I first show that this s is sufficient and then that it is minimally sufficient.

Note that this statistic s produces a partition of the sample space $A = \{A_c : \exists\ \mathbf{x} \in \mathcal{X},\ s(\mathbf{x}) = c\}$. For each set A_c of the partition A fix a point $\mathbf{x}_c \in \mathcal{X}$ to represent it.

Let $\mathbf{x}$ be a sample of X and define $\mathbf{y} = \mathbf{x}_{s(\mathbf{x})}$. Note that sample $\mathbf{y}$ is a function of $s(\mathbf{x})$ only and $s(\mathbf{x}) = s(\mathbf{y})$. Consider the joint distribution of $\mathbf{x}$

$$\mathbb{P}(\mathbf{x};\theta) = \mathbb{P}(\mathbf{x};\theta)\frac{\mathbb{P}(\mathbf{y};\theta)}{\mathbb{P}(\mathbf{y};\theta)} = \mathbb{P}(\mathbf{y};\theta)\frac{\mathbb{P}(\mathbf{x};\theta)}{\mathbb{P}(\mathbf{y};\theta)}$$

By our assumptions of s, we have that $\frac{\mathbb{P}(\mathbf{x};\theta)}{\mathbb{P}(\mathbf{y};\theta)}$ is independent of θ. Thus, we can produce the following decomposition

$$\begin{aligned} \mathbb{P}(\mathbf{x};\theta) &= h(\mathbf{x})g(s(\mathbf{x});\theta) \\ \text{where} & \\ h(\mathbf{x}) &= \tfrac{\mathbb{P}(\mathbf{x};\theta)}{\mathbb{P}(\mathbf{y};\theta)} \\ g(s(\mathbf{x});\theta) &= \mathbb{P}(s(\mathbf{y}); theta) \end{aligned}$$

By the Fisher-Neyman factorisation criterion we can deduce that s is sufficient.

Now, let t be another sufficient statistic for θ and let $\mathbf{x}, \mathbf{y} \in \mathcal{X}$ st $t(\mathbf{x}) = t(\mathbf{y})$. By the Fisher-Neyman factorisation criterion, we have

$$\begin{aligned} \mathbb{P}(\mathbf{x};\theta) &= h(\mathbf{x})g(t(\mathbf{x});\theta) \\ &= \tfrac{h(\mathbf{x})}{h(\mathbf{y})}h(\mathbf{y})g(t(\mathbf{y});\theta) \\ &= \tfrac{h(\mathbf{x})}{h(\mathbf{y})}\mathbb{P}(\mathbf{y};\theta) \text{ by Fisher-Neyman factorisation} \\ \implies \tfrac{\mathbb{P}(\mathbf{x};\theta)}{\mathbb{P}(\mathbf{y};\theta)} &= \tfrac{h(\mathbf{x})}{h(\mathbf{y})} \end{aligned}$$

This shows that $\frac{\mathbb{P}(\mathbf{x};\theta)}{\mathbb{P}(\mathbf{y};\theta)}$ is independent of θ, meaning $s(\mathbf{x}) = s(\mathbf{y})$ by our assumptions of s. This result means there exists a function f st $s(\mathbf{x}) = f(t(\mathbf{x}))\ \forall\ \mathbf{x} \in \mathcal{X}$. Moreover, due to the arbitrary definition of t, for each sufficient statistic of θ there exists a function which maps from it to our statistic s, fulfilling the definition of s being minimally sufficient. □

Theorem 4.2 states that if the ratio of the marginal distributions of two samples from a model are independent of the model parameters if, and only if, the samples map to the same value under some statistic s, then s is minimally sufficient. This property can be used to identify minimally sufficient summary statistics, either by assisting in deduction or by verifying a proposed statistic.

Statistics carry information about sampled data, but in Bayesian modelling most problems centre around estimating parameter values. In some cases a sufficient statistic may be a good estimator of a model parameter too, in **Example 4.1** it was shown that the sample mean is a sufficient statistic for the population mean of a normal distribution. This is not always the case, in **Example 4.2** it was shown that the sum of sampled values is a sufficient statistic for the mean of a Poisson distribution but this is not a good estimator.

Theorem 4.3 (Rao-Blackwell Theorem, Rao [1945]; Blackwell [1947])
Let X be a model with parameters θ, $U = u(X)$ be an unbiased estimator for function $g(\theta)$ and $s(X)$ is a sufficient statistic for θ.

> *The statistic $v(X) := \mathbb{E}[u|T = t(X)]$ is an unbiased estimator of $g(\theta)$ and $Var(v(X)) \leq Var(u(X))$.*

The statistic $v(X)$ is known as the Rao-Blackwell Estimator.

Proof. The proof that $v(X)$ is unbiased is immediate from the Tower Law

$$\begin{aligned}\mathbb{E}[v(X)] &= \mathbb{E}[\mathbb{E}[u|T = t(X)]] \\ &= \mathbb{E}[u] \\ &= g(\theta)\end{aligned}$$

Now consider the variance of $v(X)$

$$\begin{aligned}\mathrm{Var}(v(X)) &= \mathrm{MSE}[v(X)] - \mathrm{Bias}[v(X)]^2 = \mathrm{MSE}[v(X)] \\ &= \mathbb{E}[(v(X) - g(\theta))^2] \\ &= \mathbb{E}[(\mathbb{E}[v|T = t(X)] - g(\theta))^2] \\ &= \mathbb{E}[(\mathbb{E}[v - g(\theta)|T = t(X)])^2] \\ &\overset{[10]}{\leq} \mathbb{E}[(v - g(\theta))^2|T = t(X)] \\ &= \mathrm{Var}(u(X)) \\ \implies \mathrm{Var}(v(X)) &\leq \mathrm{Var}(u(X))\end{aligned}$$

□

$\overline{\mathrm{Var}(X) = \mathbb{E}[X^2] - \mathbb{E}[X]^2 \implies \mathbb{E}[X^2] \geq \mathbb{E}[X]^2}$

The Rao-Blackwell theorem (**Theorem 4.3**) provides a general relationship between estimators and sufficient statistics by demonstrating a transformation of an unbiased estimator, using a sufficient statistic, which produces an unbiased estimator with decreased variance and thus reduced mean-squared error. This is desirable as it is often straight-forward to derive a crude estimator and then apply this transformation in order to improve its performance. A Rao-Blackwell transformation is idempotent as applying it to an already transformed estimator returns the same estimator, the proof of this follows immediately from the Tower Law.

The Lehmann-Scheffe theorem [Lehmann and Scheffé, 1950] states that if the statistic used in a Rao-Blackwell transformation is both sufficient and complete, then the resulting estimator is in fact the unique minimum-variance unbiased-estimator. This result is independent of how good the initial estimator was.

Sufficiency In Practice

In Bayesian modelling problems we want to deduce the posterior for some model parameters to as high a degree of accuracy as possible. Let $f^*(\theta|X(\theta) = x_{obs})$ be the true posterior for model parameters θ and $\hat{f}(\theta|s(X(\theta)) = s(x_{obs}))$ be the estimated posterior produced by our modelling method, given x_{obs} was observed from the true model and summary statistics $s(\cdot)$ were used. If the summary statistics $s(\cdot)$ are sufficient then the estimated posterior $\hat{f}$ will converge towards the true posterior f^*, given enough simulations, however, if $s(\cdot)$ are not sufficient then $\hat{f}$ can never (consistently) converge on the true posterior f^*, and rather will always be an approximation. Thus, finding sufficient statistics for our models is highly desirable in Bayesian modelling.

Theorem 4.4 (Pitman–Koopman–Darmois Theorem, Andersen [1970])
Among families of probability distributions whose domain does not vary with the parameter being estimated, only in exponential families are there sufficient statistics whose dimension are bounded as the sample size increases.

Proof. See [Darmois, 1935; Pitman, 1936; Koopman, 1936] for the original proofs. □

However, although sufficient statistics do exist for all models, as the identity function is a sufficient statistic for all models, they are not necessarily the best choice of summary statistic when implementing computational methods as they may provide very little dimensionality reduction relative to other statistics which still manage to retain a large about of the relevant data from a sample. Moreover, the Pitman-Koopman-Darmois theorem **Theorem 4.4** states that sufficient summary statistics which provide a high level of dimensionality reduction only exist for probability distributions from exponential families.

This lack of computationally efficient sufficient statistics, for most models, motivated the concept of "approximate sufficiency" in [Joyce and Marjoram, 2008] which aims to balance the number of summary statistics with the amount of information being retained from a sample. I discuss this concept more when I present the summary statistic selection algorithm from [Joyce and Marjoram, 2008] in **Section 4.3.1**.

It is demonstrated in [Ruli, 2018] that the using summary statistics which are sufficient for parameters produces unreliable results when performing model selection. This is due to it being impossible to distinguish between models which have the same sufficient statistics for their parameters. For example, the sum of sampled values is a sufficient statistics for the means of both geometric and Poisson distribution and so cannot be used to compare these two models. Rather, cross-model sufficient statistics would be required to distinguish between these models in practice, which is impossible in practice.

To close this section, I shall mention the Ewens' Sampling formula Ewens [1972] which illustrates a real-world scenario where useable and useful sufficient statistics have been found. The Ewens' Sampling formula provides, under certain conditions, a parametric probability distribution for the frequencies of unique types of allele observed in a sample of gametes when using the Infinite Alleles model. The mutation rate is the only parameter of this distribution and it is notable that the total number of types is a sufficient statistic for the mutation rate [Joyce, 1998]. This is especially appealing as ABC methods are used widely in population genetics research (See [Wegmann and Excoffier, 2010; Beaumont *et al.*, 2002; Marjoram and Tavaré, 2006] among many others).

4.3 Methods for Summary Statistic Selection

When thinking about summary statistic selection it is useful to consider the summary statistics themselves as a feature of your theorised model. This makes the process of selecting summary statistics analogous to model selection, with each combination of summary statistics being considered as a unique model. This is the motivation behind many summary statistic selection methods.

4.3.1 Approximate Sufficient Subset

[Joyce and Marjoram, 2008] presents the first algorithm for automating the selection of summary statistic. The key idea of their approach is to find a subset of summary statistics, from a large set of hand-crafted statistics, such that ABC methods perform approximately as well when using the subset. This requires a method for empirically evaluating the information extracted by sets of summary statistics. The use of hand-crafted statistics, as discussed above, comes with its own advantages and limitations.

Remark 4.2 (Difference of Log-Likelihood)
Let $s_1, \ldots, s_k$ be summary statistics for a model X with parameters θ. Define sets $s_{1:k-1} := \{s_1, \ldots, s_{k-1}\}$, $s_{1:k} := \{s_1, \ldots, s_k\}$ and consider the likelihood of the set $s_{1:k}$ with respect to the model parameters θ

$$\begin{aligned} & & \mathbb{P}(s_{1:k}(X)|\theta) &= \mathbb{P}(s_k(X)|s_{1:k-1}(X),\theta) \cdot \mathbb{P}(s_{1:k-1}(X)|\theta) \\ \implies & & \ln \mathbb{P}(s_{1:k}(X)|\theta) &= \ln \mathbb{P}(s_k(X)|s_{1:k-1}(X),\theta) + \ln \mathbb{P}(s_{1:k-1}(X)|\theta) \\ \implies & \ln \mathbb{P}(s_{1:k}(X)|\theta) - \ln \mathbb{P}(s_{1:k-1}(X)|\theta) &= \ln \mathbb{P}(s_k(X)|s_{1:k-1}(X),\theta) \end{aligned}$$

For the theoretical basis of their algorithm, Joyce & Marjoram first show that the difference in log-likelihood value between two sets of summary statistics can be directly quantified as $\ln \mathbb{P}(s_k(X)|s_{1:k-1}(X),\theta)$ (**Remark 4.2**). It is worth noting that if the set $s_{1:k-1}$ is sufficient for model parameter θ then the quantity $\ln \mathbb{P}(s_k(X)|s_{1:k-1}(X),\theta)$ would be independent of θ and thus mean s_k does not contribute to inferences about θ. This result reduces the problem of comparing sets of statistics to calculating or estimating a single value and motivates Joyce & Marjoram use of log-likelihood in their definition of score. Score quantifies how much extra information is extracted when a single extra statistic is added to a set with greater score values meaning more extra information is extracted. Thus we want to find the statistics with the greatest scores. Moreover, if the score of a statistic differs significantly from 0 then it should be accepted.

Definition 4.3 (Score δ_k, Joyce and Marjoram [2008])
Let $s_1, \ldots, s_k$ be k summary statistics. The score of s_k relative to the set $s_{1:k-1} := \{s_1, \ldots, s_{k-1}\}$ is defined as

$$\delta_k := \sup_\theta \{\ln \mathbb{P}(s_k|s_{1:k-1})\} - \inf_\theta \{\ln \mathbb{P}(s_k|s_{1:k-1})\}$$

Definition 4.4 (ε-Approximate Sufficiency, Joyce and Marjoram [2008])
Let $s_1, \ldots, s_k$ be k summary statistics. The set $s_{1:k-1} := \{s_1, \ldots, s_{k-1}\}$ ε-sufficient for statistic s_k if the score of s_k relative to $s_{1:k-1}$ is no greater that ε.

$$\delta_k \leq \varepsilon$$

ABC methods are applied in scenarios where likelihoods are intractable. This means that the score of a statistic is intractable too. Thus, Joyce & Marjoram only use the score to motivate their algorithm and in practice use different approaches to compare statistics. I discuss this in more detail later when I explore the practicalities of their algorithm.

Algorithm 4.1 (Approximately Sufficient Subset of Summary Statistics, Joyce and Marjoram [2008])

require: *Set of summary statistics S; Score threshold ε*

$S' \leftarrow \emptyset$
while *true* **do**
  Calculate the score for each statistic in S wrt S'
  $\delta_{max} \leftarrow \max_{s \in S} Score(s; S')$
  $s_{max} \leftarrow argmax_{s \in S} Score(s; S')$
  if $\delta_{max} > \varepsilon$ **then** $S' \leftarrow S' \cup \{s\}$;
  else return S' ;

Joyce & Marjorams' algorithm (**Algorithm 4.1**) starts with an empty set and proceeds to, each iteration, add the summary statistic with the greatest score wrt the set of already selected statistics, until it believes that the none of the remaining unselected summary statistics extracts a significant amount of extra information about the model parameters. They define the concept of ε-approximate sufficient sets to formalise this stopping condition, with the algorithm running until the set of accepted summary statistics S' is ε-approximate sufficient for each unchosen summary statistic, individually. This makes ε is a parameter of the algorithm, with smaller values likely leading to more summary statistics being accepted as the threshold for the amount of extra information extracted by each new statistic is lower. Alternatively, we could fix or cap the number of summary statistics we want to be accepted from the superset.

As mentioned, in practice the score cannot be calculated. Joyce & Marjoram instead determined that a proposed statistic introduces significant extra information if the posterior of parameters accepted under its usage was significantly different from the posterior when it was not used. This approach, set out in **Algorithm 4.2**, consists of estimating the expected value and standard deviation for the number of occurrences of each parameter value; and then accepting the proposed statistic if any of the observed number of occurrences is more than four standard deviations away from its expected value[11]. For this approach to be computationally tractable the posterior space is discretised into M bins whose counts can be compared. When this approach is applied the stopping condition of the main algorithm is changed to be *"Stop if no proposed statistics were accepted in the last cycle"*. There are alternative stopping conditions which could be used, it is reasonable to place a cap on the number of statistics allowed to be accepted[12].

Algorithm 4.2 (Evaluate Proposed Statistic)

[11] In [Joyce and Marjoram, 2008] it is recommended to use a value of between one and four standard deviations
[12] A leave-one-out cross-validation could be used to determine the optimal number of statistics to use.

require: *Sets of accepted parameters* $\Theta_{1:k-1}, \Theta_{1:k}$*; Number of bins* M

1 $N_{1:k} \leftarrow |\Theta_{1:k}|$

2 $N_{1:k-1} \leftarrow |\Theta_{1:k-1}|$

3 $C_{1:k-1} \leftarrow \Theta_{1:k-1}$ *discretised into* M *bins*

4 $C_{1:k} \leftarrow \Theta_{1:k}$ *discretised into* M *bins*

5 $E \leftarrow \dfrac{C_{1:k-1} \cdot N_K}{N_{K-1}}$; `// Expected value of each bin`

6 $sd \leftarrow \sqrt{\dfrac{E(N_{K-1} - C_{1:k-1})}{N_{K-1}}}$; `// Standard deviation of each bin`

7 **if** *Any* $|C_{1:k} - E| > 4sd$ **then return** *Accept proposed statistic ;*

8 **else return** *Reject proposed statistic;*

The expected values E (Line 5), the standard deviations sd (Line 6) and the condition of the if statement (Line 7) are each evaluated piece-wise.

Algorithm 4.2 requires sets of parameters which were accepted under each set of summary statistics in order to compare posteriors. These sets are acquired by generating a large number of simulations of the theorised model, using parameters sampled from the model priors, and then running ABC-Rejection Sampling to determine which parameters would be accepted under each set of summary statistics[13]. This approach has the desirable property that we only need to generate simulations once, and can then use the same set of samples each time we run **Algorithm 4.2**. This property allows us to justify generating a very large number of simulations which will make the posterior estimates more accurate. Using this approach means the approximation factor ε is no longer a parameter of the algorithm, but the distance measure, acceptance kernel and bandwidth used in the ABC-Rejection Sampling step are now parameters, as well as the number of bins M and number of model simulations. Implement caching to avoid having to run ABC-Rejection Sampling multiple times for the same set of statistics will dramatically improve the computational efficiency of this approach, especially when a large super-set of statistics is being used.

A limitation of **Algorithm 4.2** is that it does not produce a numerical value which can be used to rank each proposed statistic[14], as the theoretical score would. This means we cannot choose to keep adding the highest scoring statistic, as in **Algorithm 4.1**, and instead have to consider statistics in a somewhat arbitrary order. This means that the order in which statistics are considered will affect the result of the algorithm. An imperfect solution to this is to consider statistics in a random order and whenever a statistic is accepted, consider removing each statistic which has already been chosen. Implementing this is not trivial as considerations need to be made to avoid infinite loops where the same statistics keep getting added and removed.

Algorithm 4.2 performs poorly when the supplied set of statistics include uninformative statistics. This can be seen by noticing that a summary statistic which maps to a constant will almost always produce a posterior which is significantly different from an informative set of statistics and therefore be accepted as a statistic despite.

4.3.2 Minimising Entropy

[Nunes and Balding, 2010] explores using the set of summary statistics which minimise the entropy of the approximate posterior distribution returned by an ABC-method. In the paper

[13] Considerations need to be made for how the bandwidth of the kernel scale with the number of parameters. The simplest solution is for it to scale linearly.

[14] You could compare each possible subset but this would highly inefficient as it potentially requires $\binom{K!}{2}$ executions of Algorithms 4.2, where K is the number of statistics being considered, and there is no guarantee this would produce a definitive best set, due to the complex relationships between statistics.

Nunes & Balding propose two algorithms: the first I discus in this section; and the second, a two-step approach, I discuss in section 4.3.3. Both methods consider sets of handcrafted statistics.

Definition 4.5 (Entropy H, Shannon [1948])
The entropy $H(X)$ of a probability distribution X is a measure of the information and uncertainty in distribution.

$$\begin{aligned} \textit{Discrete} \quad H(X) &:= -\sum_{x \in \mathcal{X}} \mathbb{P}(X = x) \cdot \ln \mathbb{P}(X = x) \\ \textit{Continuous} \quad H(X) &:= -\int_{\mathcal{X}} f_X(x) \cdot \ln f_X(x) dx \end{aligned}$$

where $\mathcal{X}$ is the support of distribution X.
The joint-entropy of probability distributions $X_1, \ldots, X_n$ is defined as

$$\begin{aligned} \textit{Discrete} \quad H(X_1, \ldots, X_n) &:= -\sum_{x_1 \in \mathcal{X}_1} \cdots \sum_{x_n \in \mathcal{X}_n} \mathbb{P}(x_1, \ldots, x_n) \cdot \ln \mathbb{P}(x_1, \ldots, x_n) \\ \textit{Continuous} \quad H(X_1, \ldots, X_n) &:= -\int f_{X_1,\ldots,X_n}(x_1, \ldots, x_n) \cdot \ln f_{X_1,\ldots,X_n}(x_1, \ldots, x_n) dx \ldots dx_n \end{aligned}$$

where $\mathcal{X}_i$ is the support of distribution X_i

A greater entropy value indicates a lower amount of information in the distribution, and visa-versa. This motivates approaches which seek to minimise entropy as they will in turn maximise information. Nunes & Balding's usage of entropy is equivalent to Joyce & Marjoram's usage of score, the advantage of entropy is that there are well-studied methods for estimating its value. Entropy may appear to be an equivalent measure to variance, but this is only true for unimodal distributions. Entropy measures the spread of probability mass whereas variance measures the spread of the data values. The difference can be seen by considering how the values of entropy and variance change for a bimodal distribution if the distance between the two peaks is increased; entropy will not change, whilst variance will increase.

Definition 4.6 (k^{th}-Nearest Neighbour Estimator of Entropy, Singh *et al.* [2003])
Consider a distribution X with ρ different parameters and a set of parameter values Θ which were accepted during some ABC-method, with $n = |\Theta|$. Singh et al. *[2003] define the k^{th}-nearest neighbour estimator of entropy as*

$$\hat{H} = \ln\left(\frac{\pi^{\rho/2}}{\Gamma\left(1 + \frac{\rho}{2}\right)}\right) - \frac{\Gamma'(k)}{\Gamma(k)} + \ln(n) + \frac{\rho}{n}\sum_{i=1}^{n} \ln D_{i,k}$$

where $D_{i,k}$ is the Euclidean distance between the i^{th} accepted parameter set and its k^{th} nearest neighbour and $\Gamma(\cdot)$ is the gamma function.

In the context of summary statistic selection we want to calculate the entropy of the posterior distribution of model parameters given summary statistic values. We only ever have an approximation of this distribution and thus can only estimate its entropy. For computational efficiency it is common to discretise the approximated distribution. There are many techniques for estimating the entropy of a distribution from samples, see [Beirlant *et al.*, 1997] for an overview. Due to most models of interest in Bayesian modelling having multiple parameters and thus the posterior being multivariate, Nunes & Balding suggest using the asymptoticly k^{th}-Nearest Neighbour estimator of entropy Singh *et al.* [2003] (**Definition 4.6**).

When implementing **Definition 4.6** determining the k^{th} nearest neighbour in an efficient manner is not trivial. A truncated insertion sort is a straightforward approach but has time

complexity $O(kn)$ so does not scale efficiently for large values of k. [Singh *et al.*, 2003] recommend using $k = 4$ as their experiments found that greater values of k did not decrease the root-mean square error RMSE significantly, and so were not worth the increased computational complexity.

Using the "Best Samples" version of the ABC-Rejection Sampling algorithm to acquire the approximate posterior used in **Definition 4.6** is advisable as it does not require the specification of an acceptance kernel and thus the same configuration can be used for all sets of summary statistics. Also, as the number we specify the number of simulations this step should have the same run-time each time it is called, regardless of the set of statistics being analysed, assuming that the summary statistics take trivial time to calculate.

Algorithm 4.3 (Minimum Entropy Summary Statistic Selection, Nunes and Balding [2010])
require: *Set of summary statistics* S
1 **for** $S' \in 2^S$ **do**
2 $\quad \Theta \leftarrow$*Parameter sets accepted from ABC-Rejection Sampling using* S'
3 $\quad \hat{H}_{S'} \leftarrow \hat{H}(\Theta)$
4 $S^*_{ME} \leftarrow argmin_{S' \in 2^S} \hat{H}_{S'}$
5 **return** S^*_{ME}

The first algorithm proposed by Nunes & Balding **Algorithm 4.3** is very straight-forward. It calculates the entropy for each subset of the supplied set of summary statistics S and returns whichever set has the lowest entropy. A limitation of **Algorithm 4.3** is how its computational complexity scales wrt the size of the set of supplied summary statistic S. As the for-loop (line 1) considers every subset, the computational complexity of the algorithm scales exponential with the size of S. The simplest mitigation of this is to only consider subsets whose size is in some specified range, this could be implemented adaptively. A more complex procedure would be to introduce a pruning algorithm which does evaluate sets whose subsets produce high entropy values.

The estimated entropy value for a set of statistics will vary each time due to the random nature of the parameter set Θ returned by the ABC-Rejection Sampling step (Line 2). This means the set of parameters returned by **Algorithm 4.3** will vary each time it is executed. Allowing more simulations to be performed in this step will reduce the varability in the entropy results. Alternatively, you could instead run the algorithm multiple times, keeping the number of simulations performed in line 2 relatively low, and use the results to generate a mixtures model.

Algorithm 4.3 only returns the best performing set, and no other information. It could be extended to instead return the best m sets along with their entropy values so that a mixtures model could be generated.

Algorithm 4.3 only uses entropy to evaluate the sets of summary statistics. However, as justified above, having a smaller set of statistics is preferable. This preference can be encoded into the algorithm by inflating the entropy value of larger sets. How much the value should be inflated is not a trivial matter.

As each subset is assessed independently, **Algorithm 4.3** can be readily implemented using parallelisation. This will dramatically improve run time for this algorithm and is not something which can be done with Joyce & Marjorams' approximately sufficient subset approach.

4.3.3 Two-Step Minimum Entropy

The second algorithm in [Nunes and Balding, 2010] is an extension of the first. It uses the set of statistics S^*_{ME} returned by **Algorithm 4.3** to simulate parameter sets Θ_{acc} which are treated

as if they were observed. Each subset of statistics is then reassessed using these parameter sets Θ_{acc}, with the subset which optimises some error measure returned as the recommended set.

Definition 4.7 (Mean Residual Sum of Squares Error, Nunes and Balding [2010])
Let $\mathbf{X} := \{X_1, \ldots, X_n\}$ *be a set of observations and* X^* *be a target value. Residual sum of squares error (RSSE) measures the difference between the observed values and the target value by calculating the mean of the square of the residuals. A smaller RSSE value indicates less error as the observed values do not deviate much from the target value.*

$$RSSE(\mathbf{X}, X^*) := \sqrt{\frac{1}{n}\sum_{i=1}^{n} \|X_i - X^*\|}$$

where $\|\cdot\|$ *is the Euclidean distance.*

Now define $\mathbf{X}^* := \{X_1^*, \ldots, X_m^*\}$ *to be a set of target values. The mean residual sum of squares error (MRSSE) is the mean RSSE value for each target value wrt the observed data* X*.*

$$MRSSE(\mathbf{X}, \mathbf{X}^*) := \frac{1}{m}\sum_{i=1}^{m} RSSE(\mathbf{X}, X_i^*)$$

The accepted parameter sets Θ_{acc} are treated as if they are the true parameter space distribution, this means the reassessments now considers the error between a simulated distribution and Θ_{acc}. There are various measures which could be used, including Kolmogorov–Smirnov statistic [Chakravarti *et al.*, 1967] and cross-entropy. Nunes & Balding choose to use the mean residual sum of squares error (MRSSE, **Definition 4.7**) with the set of statistics which minimises MRSSE wrt Θ_{acc} is return as the recommended set of statistics.

MRSSE is a desirable statistic to use in the context of Bayesian modelling as there are theoretical results which prove that minimising MRSSE is a good metric for estimating the mean of a distribution and that posterior means are optimal summary statistics. MRSSE is straightforward to compute and can be applied to multivariate distributions is sensitive to outlier values. Note that the scale of parameter values will affect the MRSSE and thus parameter values should be standardised before computation. A limitation of MRSSE is its sensitivity of outlier values, which is not mitigated by the standardisation.

Algorithm 4.4 (Two-Step ME Summary Statistic SelectionNunes and Balding [2010])
require: *Observations from true model* x_{obs}*, Set of summary statistics* S*, Number of simulations to run* n_{run}*, Number of simulations to accept* n_{obs}

```
S_ME ← Algorithm 4.3(S)
Θ̂_ME ← Parameter sets accepted from "Best Samples" ABC-RS(x_obs, S', n_run, n_acc)
Standardise Θ̂_ME
for S' ∈ 2^S do
    Θ_acc ← Parameter sets accepted from "Best Samples"
      ABC-RS(x_obs, S', n_run, n_acc)
    Standardise Θ_acc
    MRSSE_S' ← MRSSE(Θ_acc, Θ̂_ME,i)
S* ← argmin_{S'∈2^S} MRSSE_S'
return S*
```

Algorithm 4.4 inherits many of the limitations of the **Algorithm 4.3**, namely those concerning how its performance scales with the size of S and the use of minimum entropy. The

mitigations for these are the same as discussed in Section 4.3.2. Additionally, to reduce the number of subsets being evaluated in the for-loop (line 4). As **Algorithm 4.4** requires the running of **Algorithm 4.3** it will always have greater computational complexity.

4.3.4 Semi-Automatic ABC

[Fearnhead and Prangle, 2011] presents the first algorithm which constructs its own summary statistics for ABC, rather than choose from a set of hand-crafted ones. Their approach (**Algorithm 4.5**) uses a pilot run of an ABC-method to generate a naïve approximation of the parameter posterior which is used to generate summary statistics. The approximate posterior is used to generate a "training set" from which a regression model can be fitted. Model parameters are assumed to be independent and one summary statistic is generated per each model parameter. The generated summary statistics target the posterior mean, an optimal summary statistic, and should be used in a proper running of ABC to generate parameter posteriors. This approach is referred to as semi-automatic as it requires the user to specify the summary statistics used in the pilot run of ABC however the identity function would be appropriate, although inefficient.

Algorithm 4.5 (Semi-Automatic ABC, Fearnhead and Prangle [2011])
require: *Observations from true model x_{obs}, Set of summary statistics S, Number of simulated parameter sets m, Theorised model X*
1 $f_\theta \leftarrow$ *Posterior from pilot run of an ABC-method using x_{obs} and S*
2 $\hat{\Theta} \leftarrow$ *m simulations from f_θ*
3 $X_{\hat{\theta}} \leftarrow X\left(\hat{\theta}\right)$ *for each* $\hat{\theta} \in \hat{\Theta}$
4 *Generate summary statistics using* $\hat{\Theta}$ *and* $\{X_{\hat{\theta}}\}_{\hat{\theta}\in\hat{\Theta}}$

Regression methods are used in line 4 with the goal of creating mappings from the simulated response data $x_{\hat{\theta}}$ and the generated parameter values $\hat{\Theta}$. The best regression methods are those which target the expected value of the parameter as the posterior mean is an optimal summary statistic. There are several approaches which can be taken, I outline three here

1. Linear regression [Fearnhead and Prangle, 2011] assumes that the model can be expressed as $\mathbf{y} = \alpha + \boldsymbol{\beta}^T X + \varepsilon$ where X is the explanatory variables, $\mathbf{y}$ is the response variables[15], $\alpha \in \mathbb{R}, \beta \in \mathbb{R}^{|\theta|}$ are coefficients to be fitted and ε is some zero-mean additive noise which can be modelled by a random variable. Linear regression seeks to find the values $\hat{\alpha}, \hat{\boldsymbol{\beta}}$ which optimises some loss function

$$\begin{aligned}\hat{\alpha}, \hat{\boldsymbol{\beta}} &= \text{argmin}_{\alpha,\boldsymbol{\beta}} \sum_i L\left(\mathbb{E}[y|\mathbf{x}_i, \alpha, \boldsymbol{\beta}] - y_i\right) \\ &= \text{argmin}_{\alpha,\boldsymbol{\beta}} \sum_i L\left(\alpha + \boldsymbol{\beta}^T \mathbf{x}_i - y_i\right)\end{aligned}$$

   Linear regression works well when each response variable is independent and can easily be extended to projections of X by replacing all X terms with $f(X)$ where $f(\cdot)$ is a (potentially non-linear) function. This is useful in the context of ABC-methods as we can define $f(\cdot)$ to be our summary statistics.

   Linear regression is a well study problem and there any many tractable solutions with least-squares estimation being perhaps the most popular. In ordinary least-squares estimation

[15] In Bayesian modelling context typically X is set to the observed values x_{obs} and y are set to the model parameters θ.

the quadratic loss function L_2 is used meaning the problem is to find

$$\begin{aligned}\hat{\alpha}_{LSE}, \hat{\boldsymbol{\beta}}_{LSE} &= \text{argmin}_{\alpha,\boldsymbol{\beta}} \sum_i \left(\alpha + \boldsymbol{\beta}^T \mathbf{x}_i - y_i\right)^2 \\ &= \text{argmin}_{\alpha,\boldsymbol{\beta}} \sum_i \left(\alpha + \boldsymbol{\beta}\mathbf{x}_i - y_i\right)^T \left(\alpha + \boldsymbol{\beta}^T \mathbf{x}_i - y_i\right)\end{aligned}$$

A closed-form estimator for these quantities is known [Hayashi, 2000].

$$(\hat{\alpha}_{LSE}, \hat{\boldsymbol{\beta}}_{LSE}) = \left(\tilde{X}^T \tilde{X}\right)^{-1} \tilde{X}^T \mathbf{y}$$

where $\tilde{X}$ is X with a column of 1s at the start for the constant term. There are extensions of ordinary least-squares which allow for weighting of variables and for the model to be heteroscedasticity. These extensions are not relevant to the problems being covered in this project.

2. Lasso regression [Hastie *et al.*, 2009] seeks the vector $\hat{\beta}$ which satisfies the following expression

$$\begin{aligned}\hat{\beta} &= \text{argmin}_\beta \sum_{i=1}^{N} \left(y_i - \beta_0 - \sum_{j=1}^{\rho} x_{ij}\beta_j\right)^2 \\ \text{subject to} \quad & \sum_{j=1}^{\rho} |\beta_j| \leq t\end{aligned}$$

where X are the explanatory variable values, $\mathbf{y}$ are the response variable values, $\rho := |X_i|$ is the number of model parameters and t is a restriction on the size of regression coefficients.

Lasso and Ridge regression have the same objective function, but ridge regression uses an L_2 penalty function rather than lasso's L_1 function. An L_1 penalty function is preferable for feature selection as it shrinks coefficient values to zero more aggressively than an L_2 function, this is useful if the coefficient for a feature is (near) zero then the feature can be dropped.

3. Canonical correlation analysis (CCA) [Mardia *et al.*, 1979] splits variables into two sets $\mathbf{X}, \mathbf{Y}$[16] and basis vectors $\boldsymbol{\alpha}, \boldsymbol{\beta}$ are sought such that the linear combinations $\psi := \boldsymbol{\alpha}^T \mathbf{X}$, $\phi := \boldsymbol{\beta}^T \mathbf{Y}$ are as correlated as possible.

$$\boldsymbol{\alpha}, \boldsymbol{\beta} = \text{argmax}_{\boldsymbol{\alpha},\boldsymbol{\beta}} \text{Corr}(\boldsymbol{\alpha}\mathbf{X}, \boldsymbol{\beta}\mathbf{Y})$$

Solutions to this are known and readily calculatable.

$$\begin{aligned}\boldsymbol{\alpha} &= \Sigma_{XX}^{-1}\Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX} \\ \boldsymbol{\beta} &= \Sigma_{YY}^{-1}\Sigma_{YX}\Sigma_{XX}^{-1}\Sigma_{XY}\end{aligned}$$

where Σ_{UV} is the cross-covariance matrix of random vectors U, V. R provides an inbuilt function `cancor`.

As Lasso uses the L_1 penalty function, which is non-linear, there is no closed expression of Lasso regression. Meaning that computing a solution to Lasso has $O(N^2)$ time-complexity[17]. Fearnhead & Prangle recommend the use of linear regression as it is straight-forward to implement and does not perform notably worse than the other approaches in general.

[16] For Bayesian modelling you typically set $\mathbf{X}$ to be the model parameters and $\mathbf{Y}$ to be observed values.
[17]

For their specific implementation of linear least-squares regression they treat each model parameter θ_i completely separately and allow for mappings $f(\cdot)$ of the response data. This means they are fitting $\rho = |\theta|$ different models

$$\theta_i = \alpha^{(i)} + (\boldsymbol{\beta}^{(i)})^T f(\mathbf{x}) + \varepsilon_i$$

As ABC-methods only consider the distance between summary statistic values, the constant terms $\alpha^{(i)}$ can be neglect from our generate summary statistics. This means the summary statistic s_i for the i^{th} model parameter is defined as

$$s_i(\mathbf{x}) = \hat{\beta}^{(i)} f(\mathbf{x})$$

The mapping $f(\cdot)$ is a parameter of this algorithm and should be used to encode likely relationships between observations and parameters, however it can just be set to the identity function for simplicity. As the mapping is part of the generated summary statistic s_i it is important for it to be computationally efficient, it order for the summary statistic to be efficient.

Algorithm 4.6 (Semi-Automatic ABC - Least Squares)

require: *Observations from true model x_{obs}, Set of summary statistics S, Number of simulated parameter sets m, Theorised model X, Mapping $f(\cdot)$*

1 $f_\theta \leftarrow$ *Posterior from pilot run of an ABC-method using x_{obs} and S*
2 $\hat{\Theta} \leftarrow$ *m simulations from f_θ*
3 $X_{\hat{\theta}} \leftarrow X\left(\hat{\theta}\right)$ *for each* $\hat{\theta} \in \hat{\Theta}$
4 $\hat{X} \leftarrow \{X_{\hat{\theta}_1}, \ldots, X_{\hat{\theta}_m}\}$
5 $F \leftarrow f(\hat{X})$
6 $\tilde{F} \leftarrow F$ *with a preceding column of 1s*
7 **for** $i = 1, \ldots, \rho$ **do**
8 $\quad A_i \leftarrow i^{th}$ *element of each set in* $\hat{\Theta}$
9 $\quad (\alpha^{(i)}, \boldsymbol{\beta}^{(i)}) \leftarrow (\tilde{F}^T \tilde{F}^{-1}) \tilde{F}^T A_i$
10 $\quad s_i(\mathbf{x}) := \boldsymbol{\beta}^{(i)} \mathbf{x}$
11 **return** $\{s_1, \ldots, s_\rho\}$

$\rho := |\theta|$, the number of model parameters.

Algorithm 4.6 is a restatement of the general algorithm **Algorithm 4.5** using linear least-squares regression. Any ABC-method can be used for the pilot run (Line 1), using the "Best Samples" version of ABC-Rejection Sampling is it has the simplest acceptance criteria to define and the most predictable run-time. Further, any set of summary statistics S can be used to. The pilot run is an opportunity for expert knowledge to be encoded into the model by hand-crafted statistics, but, as this algorithm will mainly be run when such statistics are not known, the identity function can be used for simplicity and guaranteed sufficiency. The closer the posterior produced by the pilot run, the more representative the generated values (lines 2-3) will be and thus the more informative the regression fit will be, creating better summary statistics. The other time expert knowledge can be encoded is in the specification of map $f(\cdot)$.

The least-squares approach used in **Algorithm 4.6** treats each model parameter as fully independent. This may not be true and ignoring this may lead to missed insights. Different regression approaches can be used to maintain dependencies between parameters (e.g. CCA). The generated summary statistics offer little insight or interpretability, on their own, but can be viewed intuitively as posterior mean estimators due to how they generated. This approach generates one summary statistic for each model parameter, if it could incorporate dependencies

between model parameters then the total number of summary statistics could be reduced, increasing the compression level.

Using the generated summary statistics in ABC-methods is not straightforward as we lack the intuition required to defined acceptance criteria. The use of adaptable versions of the ABC-methods avoids this issue as you only have to specify what acceptance rate you wish to achieve.

4.3.5 Non-Linear Projection

The semi-automatic approach of [Fearnhead and Prangle, 2011] does allow for non-linear projections from the response data x to the parameter values θ but the user needs to specify the non-linear functions. More specifically, **Algorithm 4.6** produces non-linear projections if, and only if, the mapping $f(\cdot)$ is non-linear.

Being able to generate non-linear projections is desirable as it is not guaranteed that an (accurate) linear projection from response variables to model parameters exists. [Wong *et al.*, 2018] presents the first attempt at using a deep neural-network[18] to construct summary statistics. The general approach to ABC is the same as [Fearnhead and Prangle, 2011]: Perform a pilot run to generate training data; Train a neural network to fit response values to parameter values; And, then use the trained network to calculate summary statistic values for a proper run of ABC. Due to the flexibility of DNNs the number of outputs (i.e. the dimensions of the summary statistic) can be specified to any value, although more outputs require more training time and potentially a larger network.

The network used to demonstrate this approach in [Wong *et al.*, 2018] is fairly small with three hidden layers, with 5-5-3 nodes each, and takes all the observed data as an input. The model was trained to fit to parameter values, resulting in summary statistics which approximate the posterior mean. They demonstrate their method on an Ising model and moving-average model and show it to outperform the usage of hand-crafted summary statistics and semi-automatic ABC. The trade-off is that their DNN approach requires significantly more time than the other approaches, requiring twenty minutes when Fearnhead & Prangle's semi-automatic approach required less than one.

A natural extension to this approach is to apply a mapping to the observed data before it is passed to the network, as in semi-automatic ABC. This would allow for the encoding of expert knowledge into the network which would mean a smaller network is required, reducing training time.

This use of a neural network is liable to the same issues as many other uses, with the most dangerous being overfitting. Overfitting occurs when a neural network models the training data too closely and therefore does not perform well with more general data. Early stopping and regularisation are standard practices to mitigating overfitting. Additionally, improving the training set can help too. The training set can be improved by increasing its size and its diversity so that it is more representative of the general space. In this particular context, as the training set is generated from a posterior from a pilot run of ABC, we can improve the quality of the training set by improving this posterior. Allowing the pilot run to complete more simulations is guaranteed to improve the posterior, especially when using the identity function as the only summary statistic (due to the sufficiency of the identity function). Alternatively, the use of less naïve statistic will help too but it can be hard to identify these in practice. Using neural networks offers no interpretability of what inferences are being made, without very intricate investigation of the network.

[18] A feedforward neural-network is presented too, but these cannot model non-linear relationships unless they use a non-linear activation function.

4.3.6 Toy Example

Using identity function may not be ideal as it seeks to minimise total loss and doesn'y prioritise any features of the response. For some problems we want to prioritise features of the outcome (e.g. peak infection date in SIR model).

4.4 Model Selection

Theorems which state when a model is misspecified that bayesian inference will put mass on the distributions "closest to the ground truth" rely on strong regularity conditions. [Grünwald and van Ommen, 2018]

Introduce learning rate (SafeBayes) [Grünwald and van Ommen, 2018]

5 ABC and Epidemic Events

5.1 SIR Model

6 Conclusion

6.1 Future Areas of Research

Bibliography

Andersen, E. B. (1970). Sufficiency and exponential families for discrete sample spaces. *Journal of the American Statistical Association* **65**(331), 1248–1255.

Balakrishnan, S. (2019). Lecture notes in 36-705: Intermediate statistics, lecture 12. http://www.stat.cmu.edu/ siva/705/lec12.pdf.

Beaumont, M. A. (2019). Approximate bayesian computation. *Annual Review of Statistics and Its Application* **6**(1), 379–403.

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

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

Beirlant, J., Dudewicz, E., Györfi, L. and Dénes, I. (1997). Nonparametric entropy estimation. an overview. *INTERNATIONAL JOURNAL OF MATHEMATICAL AND STATISTICAL SCIENCES* **6**(1), 17–39.

Bellman, R. E. (1961). *Adaptive Control Processes: A Guided Tour*. First introduction of curse of dimensionality.

Beyer, K., Goldstein, J., Ramakrishnan, R. and Shaft, U. (1999). When is "nearest neighbor" meaningful? , 217–235.

Blackwell, D. (1947). Conditional Expectation and Unbiased Sequential Estimation. *The Annals of Mathematical Statistics* **18**(1), 105 – 110.

Burr, T. and Skurikhin, A. (2013). Selecting summary statistics in approximate bayesian computation for calibrating stochastic models. *BioMed research international* **2013**, 210646.

Casella, G. and Berger, R. (2001). *Statistical Inference*. Duxbury Resource Center.

Chakravarti, I., Laha, R. and Roy, J. (1967). Handbook of methods of applied statistics (v. 1), 392–394.

Darmois, G. (1935). Sur les lois de probabilité à estimation exhaustive. *Comptes Rendus de l'Académie des Sciences* , 1265–1266.

Del Moral, P. (1997). Nonlinear filtering: Interacting particle resolution. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics* **325**(6), 653–658.

Didelot, X., Everitt, R. G., Johansen, A. M. and Lawson, D. J. (2011). Likelihood-free estimation of model evidence. *Bayesian Anal.* **6**(1), 49–76.

Dodge, Y., Institute, I. S. and Commenges, D. (2006). *The Oxford Dictionary of Statistical Terms*. Oxford University Press.

Epanechnikov, V. A. (1969). Non-parametric estimation of a multivariate probability density. *Theory of Probability & Its Applications* **14**(1), 153–158.

Ewens, W. (1972). The sampling theory of selectively neutral alleles. *Theoretical Population Biology* **3**(1), 87–112.

Fearnhead, P. and Prangle, D. (2011). Constructing summary statistics for approximate bayesian computation: Semi-automatic abc .

Filippi, S., Barnes, C., Cornebise, J. and Stumpf, M. P. H. (2012). On optimality of kernels for approximate bayesian computation using sequential monte carlo .

Fisher, R. A. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London, A* **222**, 309–368.

Gelman, A. and Rubin, D. B. (1992). Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science* **7**(4), 457 – 472.

Gelman, A., Gilks, W. R. and Roberts, G. O. (1997). Weak convergence and optimal scaling of random walk Metropolisalgorithms. *The Annals of Applied Probability* **7**(1), 110 – 120.

Grelaud, A., Marin, J.-M., Robert, C. P., Rodolphe, F. and Taly, J.-F. (2009). ABC likelihood-free methods for model choice in Gibbs random fields. *Bayesian Analysis* **4**(2), 317 – 335.

Grünwald, P. and van Ommen, T. (2018). Inconsistency of bayesian inference for misspecified linear models, and a proposal for repairing it .

Hastie, T., Tibshirani, R. and Friedman, J. (2009). *The elements of statistical learning: data mining, inference and prediction*. Springer, 2nd edition.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**(1), 97–109.

Hayashi, F. (2000). *Econometrics*. Princeton Univ. Press, Princeton, NJ [u.a.].

Hinneburg, A., Aggarwal, C. C. and Keim, D. A. (2000). What is the nearest neighbor in high dimensional spaces? , 506–515.

Jasra, A., Holmes, C. C. and Stephens, D. A. (2005). Markov Chain Monte Carlo Methods and the Label Switching Problem in Bayesian Mixture Modeling. *Statistical Science* **20**(1), 50 – 67.

Jeffreys, H. (1961). *Theory of Probability*. Oxford, Oxford, England, 3rd edition.

Joyce, P. (1998). Partition structures and sufficient statistics. *Journal of Applied Probability* **35**(3), 622–632.

Joyce, P. and Marjoram, P. (2008). Approximately Sufficient Statistics and Bayesian Computation. *Statistical Applications in Genetics and Molecular Biology* **7**(1), 1–18.

Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association* **90**(430), 773–795.

Koopman, B. O. (1936). On Distributions Admitting a Sufficient Statistic. *Transactions of the American Mathematical Society* **39**(3).

Lehmann, E. L. and Scheffé, H. (1950). Completeness, similar regions, and unbiased estimation: Part i. *Sankhyā: The Indian Journal of Statistics (1933-1960)* **10**(4), 305–340.

Mardia, K., Kent, J. and Bibby, J. (1979). *Multivariate analysis*. Probability and mathematical statistics, Acad. Press, London [u.a.].

Marjoram, P. and Tavaré, S. (2006). Modern computational approaches for analysing molecular genetic variation data. *Nat Rev Genet* **7**, 759–770.

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.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics* **21**(6), 1087–1092.

Neyman, J. (1935). Sur un teorema concernente le cosidette statistiche sufficienti. *Giorn. Ist. Ital. Att., 6* , 320–334.

Nunes, M. and Balding, D. (2010). On optimal selection of summary statistics for approximate bayesian computation. *Statistical Applications in Genetics and Molecular Biology* **9**(1).

Pitman, E. J. G. (1936). Sufficient statistics and intrinsic accuracy. *Proceedings of the Cambridge Philosophical Society* , 567–579.

Pritchard, J., Seielstad, M., Perez-Lezaun, A. and Feldman, M. (1999). Population growth of human y chromosomes: A study of y chromosome microsatellites. *Molecular biology and evolution* **16**, 1791–8.

Radovanovic;, M., Nanopoulos;, A. and Ivanovic, M. (2010). Hubs in space: Popular nearest neighbors in high-dimensional data. *Journal of Machine Learning Research* **11**(86), 2487–2531.

Rao, C. R. (1945). *Information and accuracy attainable in the estimation of statistical parameters*. Bulletin of the Calcutta Mathematical Society, 81–91.

Roberts, G. O. and Rosenthal, J. S. (2001). Optimal scaling for various Metropolis-Hastings algorithms. *Statistical Science* **16**(4), 351 – 367.

Roussas, G. (1998). *A Course in Mathematical Statistics*. Academic Press, 2nd edition, 263.

Ruli, E. (2018). On model selection with summary statistics.

Schnitzer, D., Flexer, A. and Tomasev, N. (2014). Choosing the metric in high-dimensional spaces based on hub analysis. .

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal* **27**(3), 379–423.

Singh, H., Misra, N., Hnizdo, V., Fedorowicz, A. and Demchuk, E. (2003). Nearest neighbor estimates of entropy. *American Journal of Mathematical and Management Sciences* **23**(3-4), 301–321.

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.

Sisson, S. A., Fan, Y. and Beaumont, M. A. (2018). Overview of approximate bayesian computation .

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

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

Tsybakov, A. B. (2008). *Introduction to Nonparametric Estimation*. Springer Publishing Company, Incorporated, 1st edition.

Wegmann, D. and Excoffier, L. (2010). Bayesian Inference of the Demographic History of Chimpanzees. *Molecular Biology and Evolution* **27**(6), 1425–1435.

Wong, W., Jiang, B., Wu, T.-y. and Zheng, C. (2018). Learning summary statistic for approximate bayesian computation via deep neural network. *Statistica Sinica* .

Zambom, A. Z. and Dias, R. (2012). A review of kernel density estimation with applications to econometrics .