M5MS04 - Computational Statistics; Coursework 5 - Autumn 2018

Question 1. Write an essay on Approximate Bayesian Computations (ABC) of at most 3 pages, explaining it in your own words. You should address at least the following points:

- *How does it work?*
- *In what situations is it useful?*
- *What is the role of summary statistics and why are they needed?*

The following response draws heavily upon the content of reference [5].

Approximate Bayesian Computations are a class of algorithms used to draw samples from a posterior distribution $p(\theta|\mathcal{D})$ in the context of computational Bayesian inference. The basic ABC algorithm works as follows:

1. Draw a parameter value θ from the prior distribution $p(\theta)$.
2. Simulate a data set D_θ using a model parameterized by θ.
3. If the simulated data set D_θ is too different from the observed data set $\mathcal{D}$, discard θ. Otherwise, retain it.

This sampling and acceptance process repeats until a large number of parameter values have been successfully sampled. The statistic measuring the difference between two data sets is a distance measure ρ. θ is accepted if $\rho(\mathcal{D}, D_\theta) \leq \epsilon$, where $\epsilon \geq 0$ is our acceptance threshold.

In practice, $\mathcal{D}$ may have large n and/or large p. This has two consequences: first, it may be computationally expensive to evaluate the distance $\rho(\mathcal{D}, D)$. Second, the probability of generating a data set that is within a given distance of $\mathcal{D}$ will be diminished, even when sampling from the data-generating process itself[1]. Raising the acceptance threshold compromises the quality of the posterior approximation, but a low acceptance probability means the algorithm is an inefficient sampler. To circumvent these difficulties, a lower-dimensional statistic S can be used to summarize the data set. We then accept θ if $\rho(S(\mathcal{D}), S(D)) \leq \epsilon$. We will see that if S is a sufficient statistic for θ, then this acceptance criterion does not introduce additional error in approximating the posterior distribution beyond that caused by setting $\epsilon \neq 0$.

There are several properties of the ABC algorithm that make it relevant to modern applied statistics. The algorithm does not involve sampling from or evaluating a likelihood function. This is desirable if an analytical expression for the likelihood is unavailable or the likelihood is expensive to evaluate or sample from. Another, less trumpeted, property of the ABC algorithm is that the prior probability does not need to be evaluated, only sampled from. An example of where this may be the case is when the posterior of one analysis is to be used as the prior of another, but that posterior is only available in the form of a set of samples. Other noteworthy properties of the ABC algorithm are that its samples are independent and, vis-à-vis Markov-chain methods, there is no requirement for a 'burn-in' period. Since proposals are independent of one another the algorithm is easily parallelized.

A further novel property of the ABC algorithm is that it is straightforward to perform inference over multiple types of parametric model. The only adjustment necessary is that both a model and a suitable parameter vector are sampled from the prior. This suggests that for some problems, ABC may be less labour-intensive than reversible-jump Markov-chain Monte Carlo[2].

Consider the influence of the acceptance threshold ϵ on the accuracy of the ABC posterior approximation. A more restrictive acceptance tolerance will result in a more accurate posterior distribution but a less efficient sampling algorithm. If $\epsilon = 0$, then only parameterizations with non-zero probability of producing the observed data set are accepted, with those that are more likely to generate that data set being accepted more frequently. By contrast, setting $\epsilon = +\infty$ would result in accepting all proposals, making the posterior approximation identical to the prior distribution. In between these

[1] See pages 22-26 of [4] for examples of this phenomenon.

[2] The labour referred to here is deriving Jacobian matrices and implementing an arguably more involved algorithm.

two extremes, a finite non-zero threshold $\epsilon \neq 0$ would permit parameterizations which generate nearby data sets with high probability, even if they may be unlikely to generate the observed data set. A pathological example of this is the case where a parameterization has zero probability of generating the observed data set $\mathcal{D}$ but is likely to generate data sets within a a distance ϵ of $\mathcal{D}$. The ABC posteriors for $\epsilon \neq 0$ ABC approximation are a mix between the prior and the true posterior, an idea that will be explored in detail in the next question.

Judicious choice of summary statistic is essential if an ABC approximation is to both be efficient and accurately approximate the true posterior. Recall that a sufficient statistic $S_*(X)$ for a collection of independent and identically distributed random variables $X = (X_1, X_2, ..., X_n)$ allows their joint density function $f_X(x; \theta_0)$ to be factorised into the form

$$f_X(x; \theta_0) = g(x)h(\theta_0, S_*(x)) \tag{1}$$

Assume that we do not know θ_0, but we do know that f_X is the parametric form of the likelihood function. A simple piece of algebra shows that the Bayesian posterior distribution over θ is only a function of the data x by proxy of the sufficient statistic $S_*(x)$. The implication of this is that, conditional on $S_*(X)$, the posterior probability of θ is independent of X. We can therefore swap the data set $\mathcal{D}$ for its sufficient statistic $S_*(\mathcal{D})$ without affecting the posterior distribution arrived at by exact Bayesian inference.

A sufficient statistic S_* may be lower-dimensional than the data set it takes as an argument, $\mathcal{D}$. In the ABC algorithm a lower-dimensional representation of the data set is desirable for evaluating proposal acceptance. This is because fewer proposals will be necessary to obtain a given posterior approximation error[3]. Sufficient or approximately sufficient summary statistics are therefore necessary for the ABC method to be practically feasible. In some cases a low-dimensional sufficient statistic may be known, such as when the likelihood is a member of the exponential family. More often, a sufficient statistic will not be known, either because there is not an analytical expression for the likelihood available or the available expression cannot be factorised into the form of Equation 1. It then becomes necessary to use a non-sufficient summary statistic.

As is discussed in the overview paper provided for this question, no single summary statistic will be appropriate for all problems. Instead, it is necessary to consider what attributes of the posterior distribution are of interest, what model is being used, and what data is available. The information content of a summary statistic is especially relevant. Reference [5] outlines several selection criteria for summary statistics, including how dramatically they affect the ABC posterior upon being included and to what degree the posterior associated with a given summay statistic resembles a minimum-entropy posterior approximation. The authors also mention that it is essential to carefully consider how a given summary statistic will influence both the approximation of the posterior, and model selection procedures based on the Bayes factor.

A review of dimension-reduction methods for ABC outlines three classes of technique for summary statistic selection: best-subset selection, projection, and regularization [2].

- Best-subset - evaluate and rank subsets using information criteria.
- Projection - consider a lower-dimensioal linear or nonlinear combination of a set of summary statistics.
- Regularization - similar to projection, but using ridge regression so that the regression coefficients are shrunk towards zero.

It would have been interesting to explore the performance of these techniques in Question 3, however a simpler approach was taken on account of time constraints.

[3] Reference [1] suggests that the number of accepted proposals decreases exponentially with summary statistic dimension under the optimal choice of threshold value ϵ. Theory surrounding concentration of measure seems to be useful in describing problems relating to the distance between high-dimensional random vectors.

Question 2. Describe how ABC relates to the rejection sampling algorithm and parallel tempering.

Consider how we might use the acceptance-rejection (AR) method for Bayesian inference, as the ABC algorithm is. Assume that the posterior $f(\theta|\mathcal{D})$ is bounded above by some $Cg(\theta)$:

$$f(\theta|\mathcal{D}) = \frac{\pi(\theta)f(\mathcal{D}|\theta)}{Z} \leq Cg(\theta) \tag{2}$$

where C is a constant and g is a probability mass function. A valid AR method for sampling from the posterior would be to accept a proposal θ generated by pmf $g(\theta)$ with probability:

$$\frac{\pi(\theta)f(\mathcal{D}|\theta)}{Z}\frac{1}{Cg(\theta)} \tag{3}$$

Set $g(\theta) = \pi(\theta)$ so that the method generates proposals using the prior. We then require, based on Equation 2, that C satisfies the inequality:

$$\frac{\max_\theta f(\mathcal{D}|\theta)}{Z} \leq C \tag{4}$$

The acceptance probability a for $g(\theta) = \pi(\theta)$ is therefore:

$$a = \frac{f(\mathcal{D}|\theta)}{ZC} \leq \frac{f(\mathcal{D}|\theta)}{\max_\theta f(\mathcal{D}|\theta)} \tag{5}$$

i.e. the maximal acceptance probability for θ is proportional to the likelihood of generating the dataset $\mathcal{D}$ using it. In the case of the most efficient AR sampler, for which Equation 4 holds with equality, the probability that θ is proposed and then accepted is equal to

$$\frac{\pi(\theta)f(\mathcal{D}|\theta)}{\max_\theta f(\mathcal{D}|\theta)} \tag{6}$$

This result suggests that an acceptance-rejection algorithm for the posterior can be formulated exclusively in terms of the prior and likelihood without requiring knowledge of the normalizing constant Z. The toy example provided at the end of this section corroborates this conjecture.

Now consider the ABC algorithm with threshold $\epsilon = 0$. θ is proposed according to the prior probability mass function $\pi(\theta)$. The probability that a model parameterised by this proposal produces $\mathcal{D}$ is $f(\mathcal{D}|\theta)$. A proposal is only accepted if it generates exactly the observed data set, therefore the probability of proposing and accepting θ is:

$$\pi(\theta)f(\mathcal{D}|\theta) \tag{7}$$

which is less than or equal to the acceptance probability of the acceptance-rejection method, since $\max_\theta f(\mathcal{D}|\theta) \leq 1$. This exposition suggests that the $\epsilon = 0$ ABC algorithm is guaranteed to be less efficient than the acceptance-rejection method. The trade-off, of course, is that the ABC algorithm does not require the likelihood function to be evaluated.

It is natural to consider how setting $\epsilon \neq 0$ causes the ABC algorithm to deviate from acceptance-rejection. Let $\aleph(\epsilon)$ denote the set of data sets D neighboring $\mathcal{D}$ that satisfy $\rho(\mathcal{D}, D) \leq \epsilon$. Continuing to talk in terms of probability mass functions, we see that the probability of proposing θ and generating a data set within distance ϵ of $\mathcal{D}$ is:

$$\pi(\theta) \sum_{D \in \aleph(\epsilon)} f(D|\theta) \tag{8}$$

Note that when $\epsilon = 0$, $\aleph = \{\mathcal{D}\}$ and this expression is equal to Expression 7. When $\epsilon = +\infty$, the sum is equal to one and the probability of proposal and acceptance is $\pi(\theta)$. Both of these conditions are consistent with what we observed during our discussion of the ABC algorithm in the introduction. We can also see that the acceptance-rejection method could be used to generate a posterior approximation similar to that of the ABC algorithm by substituting Expression 8 for the numerator of the AR method's acceptance probability, Expression 6.

Parallel tempering is a population Markov chain method that is used primarily to ensure adequate exploration of the support when sampling from a distribution. Similar to acceptance-rejection, the acceptance probability for parallel tempering as applied to Bayesian inference would require the likelihood function to be evaluated. This makes the set of problems it is appropriate for slightly different to that of the ABC algorithm, which is used when the likelihood cannot be evaluated. Furthermore, the degree to which the support is explored by the ABC algorithm is determined by the number of proposals generated and the efficiency of the algorithm (the prior being assumed given in either case - this will affect whether support exploration is a problem). Another notable difference between the ABC algorithm and parallel tempering is that parallel tempering produces correlated samples.

```r
# Acceptance-rejection method for Bayesian inference: a toy example
set.seed(340327)
# Prior is high-variance normal
# Likelihood is normal with known sd=1
prior_mu       <- 0
prior_var      <- 10^2
likelihood_var <- 1
n <- 10 # Number of observations
x <- rnorm(n, mean=0, sd=1) # Observations
posterior_var  <- (1/prior_var + n/likelihood_var)^(-1)
posterior_mu   <- posterior_var*((prior_mu/prior_var) +
                                   (sum(x)/likelihood_var))

# Determine maximum-likelihood value for mu
log_likelihood <- function(mu) sum(log(dnorm(x=x, mean=mu, sd=likelihood_var)))
max_log_likelihood <- -optim(0, fn=function(m) -log_likelihood(m))$value

# Run acceptance-rejection sampler
N         <- 1e5
proposals <- rnorm(N, mean=prior_mu, sd=sqrt(prior_var))
log_u <- log(runif(N))
accepted <- c()
for(prop in proposals){
  if (log(runif(1)) <= log_likelihood(prop) - max_log_likelihood){
    accepted <- c(accepted, prop)
  }
}
posterior_mu # Analytical posterior mean

## [1] 0.05363202

mean(accepted) # AR posterior sample mean

## [1] 0.05207602

posterior_var # Analytical posterior variance

## [1] 0.0999001

var(accepted) # AR posterior sample variance

## [1] 0.09799476
```

> *Question 3. Implement both an ABC algorithm as well as an MCMC algorithm on an example of your own choosing. Use appropriate diagnostics for the MCMC algorithm. For the ABC part, illustrate why summary statistics are necessary and how their choice can affect the result of the algorithm.*

We consider a simple synthetic example so that both an appropriate sufficient statistic and the exact posterior distribution are known.

St. James's Gate Brewery in Dublin is the largest brewery in Ireland and is responsible for producing approximately 1 billion litres of Guinness per year [6]. As a pillar of the Irish economy, it is essential that the brewery's product remains suitably alcoholic. The alcohol content of a stout is affected by the amount of yeast in its fermentaton chamber. To evaluate the yeast concentration in the chamber, a brewer draws a sample and pours it into a glass slide known as a haemocytometer. He then counts the number of live yeast cells in each square of the haemocytometer using a microscope[4]. He does this several times, then attempts to estimate the mean number of live yeast cells per square. From this he can derive the concentration of live yeast in the chamber.

Say that the brewer draws a sample from the chamber and observes yeast cell counts of $(x_1, x_2, ..., x_n)$, where $n = 25$ is the number of squares on the haemocytometer. He passes us this data and asks us to estimate the expected yeast cell count for a slide. We assume that an appropriate likelihood function for this data set is that of the Poisson distribution:

$$\Pr(X = x|\theta) = \frac{\theta^x e^{-\theta}}{x!} : x \in \mathbb{N} \cup \{0\}, \theta > 0 \tag{9}$$

where θ denotes the expected number of yeast cells per square. Fortuitously and unbeknownest to us, this turns out to be precisely representative of the data-generating process:

```r
set.seed(23098324)
n       <- 25 # Number of haemocytometer squares
theta_0 <- 74 # This batch has more live yeast than usual
x       <- rpois(n=n, lambda=theta_0) # Yeast cell counts
```

We form a prior over θ. Our brewer friend tells us that he has no reason to think that this batch of stout will have a lower yeast concentration than previous batches, and that average yeast cell count per haemocytometer square is around 70. He mentions that the live yeast concentration is quite sensitive to the chamber conditions. In response to this information we use a Gamma prior with index $k = 10$ and scale $\omega = 7$. The prior distribution is denoted $\pi(\theta)$ and is expressed as

$$\pi(\theta) = \frac{1}{\Gamma(k)\omega^k}\theta^{k-1}e^{-\frac{\theta}{\omega}} \tag{10}$$

Since the Gamma is the conjugate prior to the Poisson distribution, the exact values of the posterior hyperparameters are:

$$k' = k + \sum_{i=1}^{n} x_i = 10 + 1869 = 1879 \tag{11}$$

$$\omega' = \frac{\omega}{n\omega + 1} = \frac{7}{25 \cdot 7 + 1} = \frac{7}{176} \tag{12}$$

The prior, posterior, and observed data are presented in Figure 1.

We evaluate the performance of a random walk Metropolis-Hastings algorithm relative to the known posterior. Throughout this section, the analytical posterior will be referred to as the target distribution. Our key measure of performance is the rate at which the Kolmogorov-Smirnov statistic converges to zero with chain length[5]. If this quantity tends to zero very quickly then the algorithm constructs an

[4] This example is based on a true story involving a real-life statistician [3].

[5] Other measures of performance could be used. The KS statistic is an unforgiving measure of approximation quality. In many cases only the mode, mean, or variance of an approximate posterior distribution would be of interest: the rate of convergence of these to their target values would all be viable alternative performance measures.

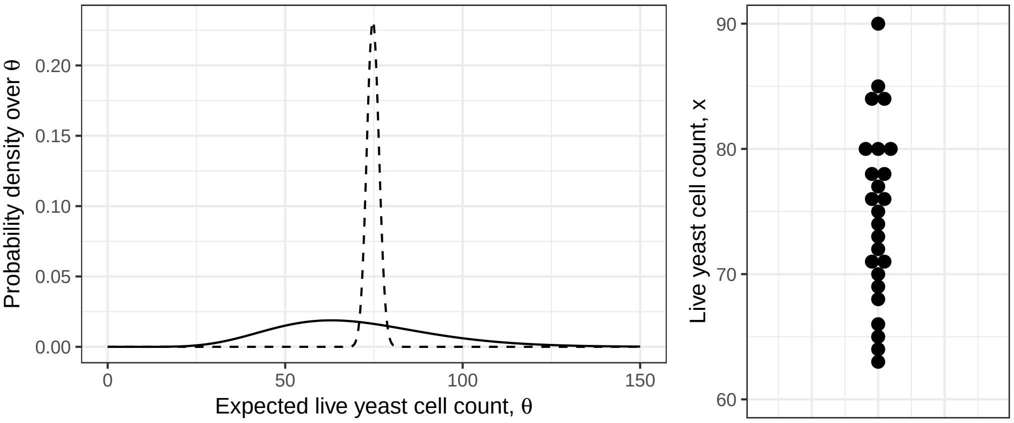

Figure 1: Q3. Left: Graph of prior probability density function (solid line) and posterior probability density function (dashed line) over parameter θ. Right: dot-plot illustrating sample distribution of yeast cell counts $(x_1, x_2, ..., x_{25})$.

accurate posterior approximation after only being run for a short while. Letting $\hat{F}_l$ denote a length-l chain's empirical distribution function (edf) and F denote the distribution function of the known Gamma posterior, this statistic is defined as:

$$K(\hat{F}_l) = ||\hat{F}_l(\theta) - F(\theta)||_\infty = \sup\ \{|\hat{F}_l(\theta) - F(\theta)| : \theta > 0\} \tag{13}$$

where the supremum is with respect to θ. This statistic is essentially the maximum difference between the empirical and target distribution functions. At each length l we will be interested in both the expected value and the variance of the KS statistic, the latter being relevant as a measure of the MH algorithm's reliability. After realizing the Markov chain M times to obtain a sequence of edfs $\hat{F}_l^{(i)} : i \in \{1, 2, ..., M\}$, we estimate these quantities using the sample mean and sample variance:

$$\mathrm{E}\, K(\hat{F}_l) \approx \quad \bar{K}(\hat{F}_l) \quad = \frac{1}{M}\sum_{i=1}^{M} K(\hat{F}_l^{(i)}) \tag{14}$$

$$\mathrm{Var}\, K(\hat{F}_l) \approx \widehat{\mathrm{Var}}\ K(\hat{F}_l) = \frac{1}{M-1}\sum_{i=1}^{M} \left(K(\hat{F}_l^{(i)}) - \bar{K}(\hat{F}_l)\right)^2 \tag{15}$$

Denote a Markov chain from the MH algorithm as $\{\Theta_1, ..., \Theta_L\}$, where Θ_j is a random variable for the parameter and L is a fixed chain length. This chain is generated by sampling an initial position θ_1 from the prior distribution $\pi(\theta)$, proposing a transition to a state θ'_2 sampled from $\mathcal{N}(\theta_1, \sigma^2)$, then accepting this proposal with probability

$$a = \min\left\{\frac{f(\theta'_2)}{f(\theta_1)},\ 1\right\}\ :\ f(\theta) = \pi(\theta)\prod_{i=1}^{n} \Pr(X = x_i|\theta) \tag{16}$$

where f is the unnormalized posterior density.

$M = 100$ length-$L = 10,000$ realizations of the Markov chain were realized. The code implementing this experiment is available at the end of this report[6]. The proposal hyperparameter was set to $\sigma = 1$, a value chosen by considering the autocorrelation sequences of chains generated across a range of σ values. A conservative burn-in period of $1,000$ was selected after inspecting the trace plots of several chain realizations.

Figure 2 plots the MCMC approximation[7] error, as measured by the KS statistic, against the chain

[6] The logarithm of the acceptance probability was taken to avoid numerical instabilities when computing the likelihood.

[7] The burn-in chain values were discarded in computing these statistics so do not form a part of the sample nor the sample length.

length, l. This chart suggests a relationship of the form

$$\log_{10} \mathrm{E}\, K(\hat{F}_l) = \beta_0 + \beta_1 \log_{10} l \tag{17}$$

$$\implies \mathrm{E}\, K(\hat{F}_l) = 10^{\beta_0} l^{\beta_1} \tag{18}$$

where $\beta_1 \approx -0.4$, $\beta_0 \approx 1.19$.

The approximation accuracy needed for the posterior is context-dependent. For the brewer's problem, we would probably come to a decision regarding the batch of stout by paying close attention to the posterior mean and variance. We will instead evaluate the quality of the posterior approximation using the KS statistic. The mean KS value for the Markov chain's empirical distribution is approximately 0.02 at $l = 10,000$, a deviation that would probably be acceptable for most problems. The 6th and 97th percentiles for the sample of KS values for chains of this length were 0.008 and 0.037 respectively, giving a gauge of best-case and worst-case error for a given chain. In a more complex problem the target distribution would not be known and it would be necessary to check that the chain distributions at convergence were consistent across a range of starting values and proposal mechanisms. Multiple-chain methods could also be used to guarantee proper exploration of the support.

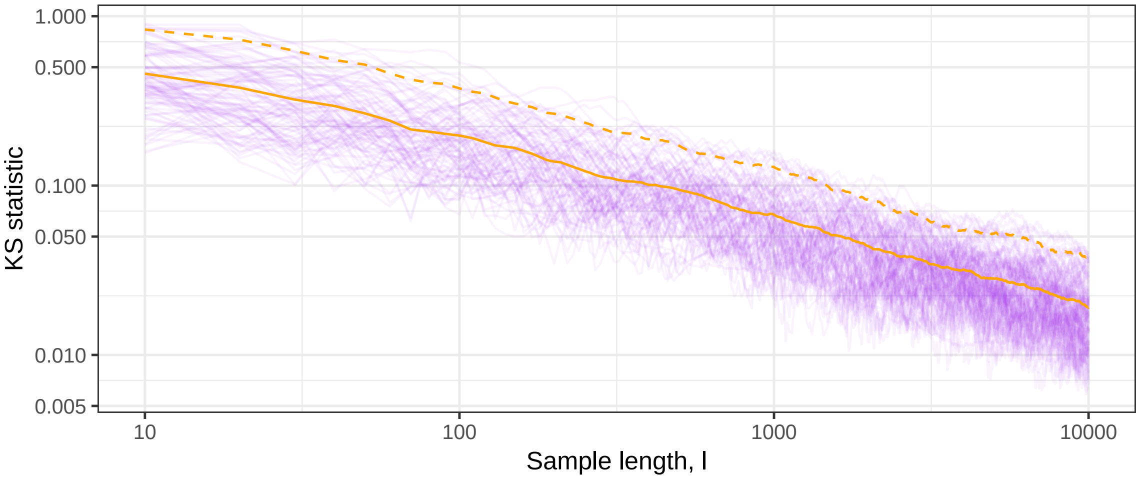

Figure 2: Q3. Approximation error of a random walk Metropolis-Hastings posterior as a function of chain length, as measured by the Kolmogorov-Smirnov statistic. Each purple line corresponds to a single chain realization ($M = 100$). The solid orange line tracks the sample mean taken across these realizations - it is an estimate of the expected KS statistic for a chain of length l. The dashed orange line is the sample mean +2 sample standard deviations of the KS statistics for that chain length.

For the ABC algorithm, the hyperparameters to be tuned are the choice of summary statistic, the acceptance threshold, and the distance measure. We use Euclidean distance. We will evaluate a selection of summary statistics relative to a known sufficient statistic for the Poisson distribution, the sum of the observations:

$$S_*(x) = \sum_{i=1}^{n} x_i \tag{19}$$

As has been discussed, this choice of summary statistic introduces no information loss (i.e. posterior approximation error) for $\epsilon = 0$. To evaluate how accurate a posterior approximation can be achieved when $\epsilon \neq 0$, the KS statistic for a 100,000-proposal ABC approximation was evaluated across equally-spaced ϵ values on a $\log_{10}$ scale. The results of this experiment are presented in Figures 3 and 4. The histograms in Figure 3 emphasise that the posterior approximation degrades with increasing ϵ: probability mass from the target posterior is squashed towards the shape of the prior. As would be expected, the approximation degrades with increasing proposal acceptance rate. Figure 4 details how the KS statistic varies as a function of both number of proposals accepted and the acceptance threshold ϵ. We can see that for $\epsilon \leq 55$ a KS statistic lower than 0.05 is achieved, but less than 6%

of proposals are accepted. There seems to be little benefit in using a more restrictive threshold than $\epsilon = 26$, since doing so does not decrease the KS statistic below an apparent saturation value of ≈ 0.03.

The sufficient statistic represents a best-case choice of summary statistic. We now evaluate the performance of the ABC algorithm for a fixed acceptance threshold and an alternative scalar summary statistic, the median of the data set. One difficulty in comparing summary statistics is that they may have different characteristic scales[8], which makes it difficult to select an acceptance threshold that does not favour a particular statistic. This is the case here, where the sum of the observations is generally in the range $1500 - 2000$ whereas the median is between 70 and 80. The median is also unusual as a summary statistic since, for count data, there is non-zero probability that a proposal's data set D satisfies $\text{median}(D) = \text{median}(\mathcal{D})$. The results of a median-based acceptance scheme are shown in Figures 5 and 6. Refer to the captions of these figures for a discussion of their results.

We close this report by demonstrating why summary statistics are necessary for high-dimensional data sets. We do this by investigating the disribution of the distance $\rho(D, \mathcal{D})$ as a function of data dimension. Figure 7 plots summary statistics for the distribution of this distance as a funciton of data set dimension. A higher threshold is necessary to achieve a given acceptance rate, but this compromises the algorithm's accuracy. Summary statistics allow us to use a low acceptance threshold while preserving an acceptable acceptance rate.

The question of whether or not to adjust the conditions for the fermentation chamber remains open - we need to discuss the posterior distribution with the brewer, who will be able to tell us whether a live yeast cell count in the range 70-80 is acceptable.

References

[1] Stuart Barber, Jochen Voss, and Mark Webster. "The rate of convergence for approximate Bayesian computation". In: *Electron. J. Statist.* 9.1 (2015), pp. 80–105. DOI: `10.1214/15-EJS988`. URL: `https://doi.org/10.1214/15-EJS988`.

[2] M. G. B. Blum et al. "A Comparative Review of Dimension Reduction Methods in Approximate Bayesian Computation". In: *Statistical Science* 28.2 (2013), pp. 189–208. ISSN: 08834237, 21688745. URL: `http://www.jstor.org/stable/43288487`.

[3] Philip Boland. "A Biographical Glimpse of William Sealy Gosset". In: *Irish Mathematical Society Newsletter* 8 (Aug. 1984). DOI: `10.1080/00031305.1984.10483195`.

[4] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning.* Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2001.

[5] Mikael Sunnr et al. "Approximate Bayesian Computation". In: *PLOS Computational Biology* 9.1 (Jan. 2013), pp. 1–10. DOI: `10.1371/journal.pcbi.1002803`. URL: `https://doi.org/10.1371/journal.pcbi.1002803`.

[6] The Irish Times. *St James's Gate brewery to expand.* 2012. URL: `https://www.irishtimes.com/news/st-james-s-gate-brewery-to-expand-1.692290` (visited on 11/12/2018).

[8] i.e. variances

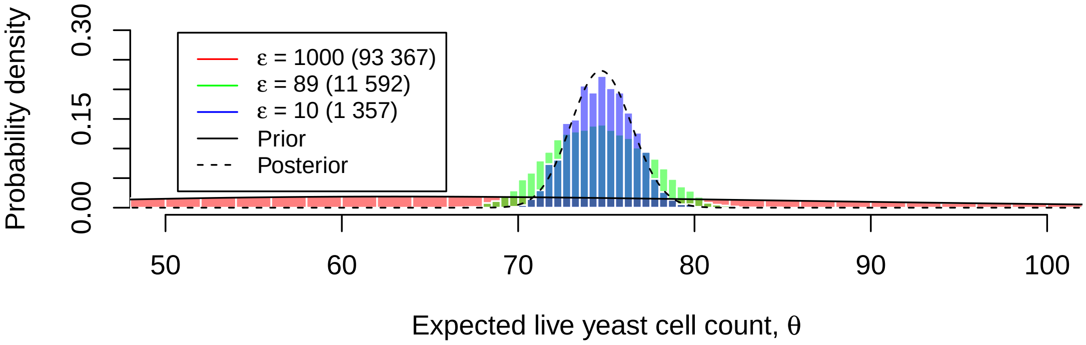

Figure 3: Q3. Sample distributions for the basic ABC algorithm. Each histogram corresponds to one $100,000$-proposal run of the algorithm. Proposals were evaluated using the sufficient statistic S_* and the specified L^2 threshold. The number of proposals that were accepted out of the $100,000$ are shown in parentheses.

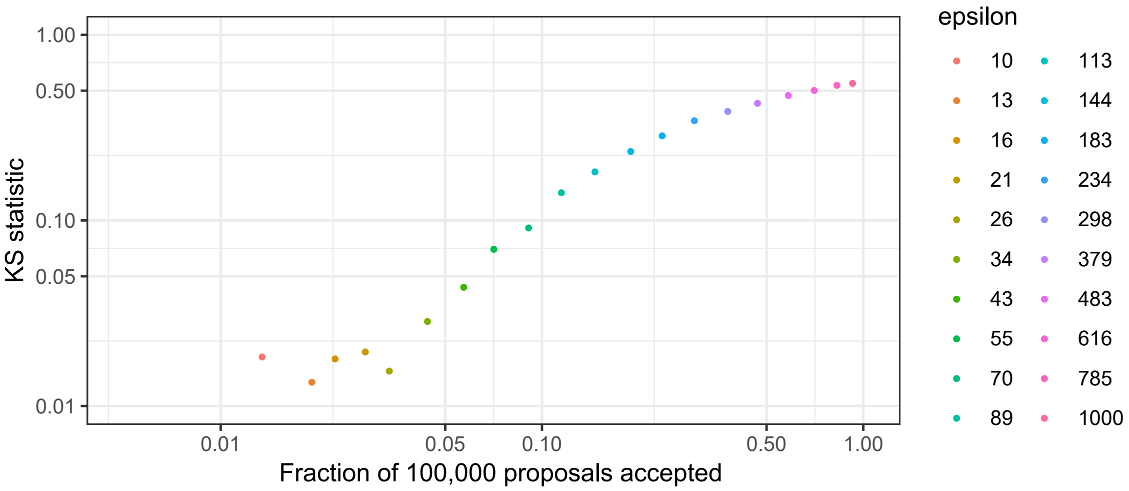

Figure 4: Q3. KS statistics for the basic ABC algorithm with a sufficient summary statistic, for ϵ between 10 and 1000. Each run of the ABC algorithm generates a set of accepted proposals of length l. The set of accepted proposals for acceptance threshold ϵ has an associated edf $\hat{F}_l^\epsilon$. Each point displayed above is a sampled value of $K(\hat{F}_l^\epsilon)$ plotted against l/N_{ABC}. A more restrictive acceptance threshold results in fewer proposals being accepted and a lower KS statistic. A laxer threshold increases the KS statistic, meaning that the ABC posterior does not resemble the target posterior.

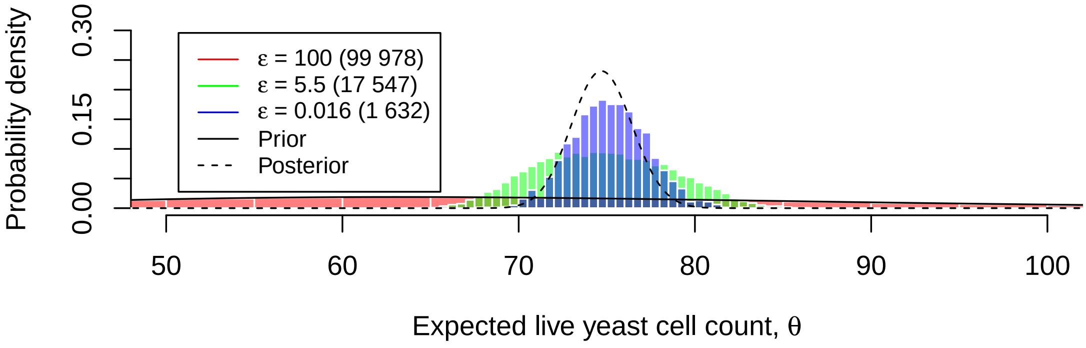

Figure 5: Q3. Sample distributions for the basic ABC algorithm using the median as a summary statistic. Each histogram corresponds to one 100,000-proposal run of the algorithm. As anticipated, using a non-sufficient summary statistic compromises the quality of the posterior approximation. The approximation's maximum density is too small, its sample variance is too large, and it is slightly off-centre relative to the target distribution. As with the sufficient statistic, raising the acceptance threshold ϵ increases the error in ABC approximation.

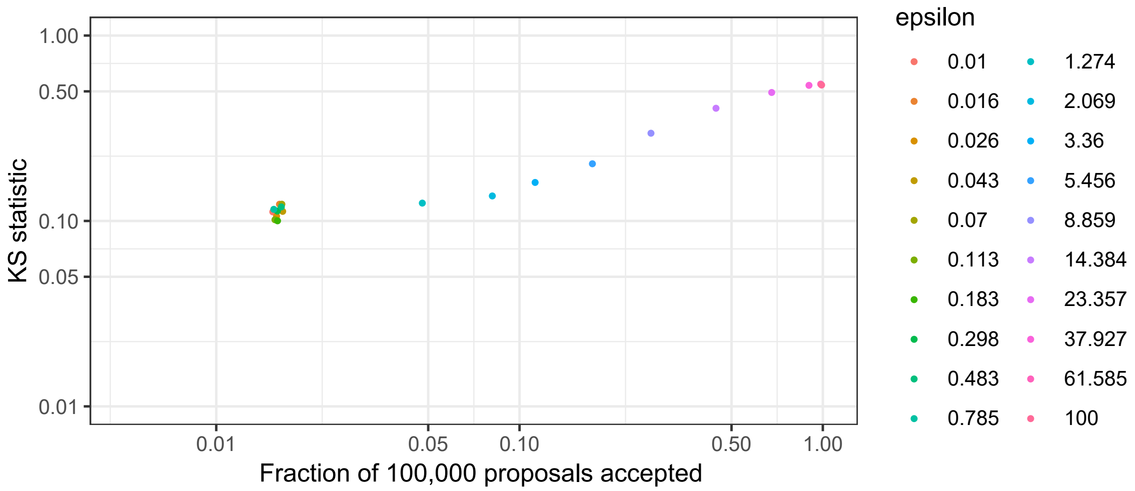

Figure 6: Q3. KS statistics for the basic ABC algorithm using the median summary statistic, for ϵ between 0.01 and 100. Each data point corresponds to a single 100,000-proposal run of the algorithm. Compare this plot with that in Figure 4: we can see that the KS statistic for a given acceptance rate is considerably higher when using the median as a summary statstic rather than the sum. We can also see that, on account of the median taking on integer values, all acceptance thresholds below 1 yield the same acceptance rate.

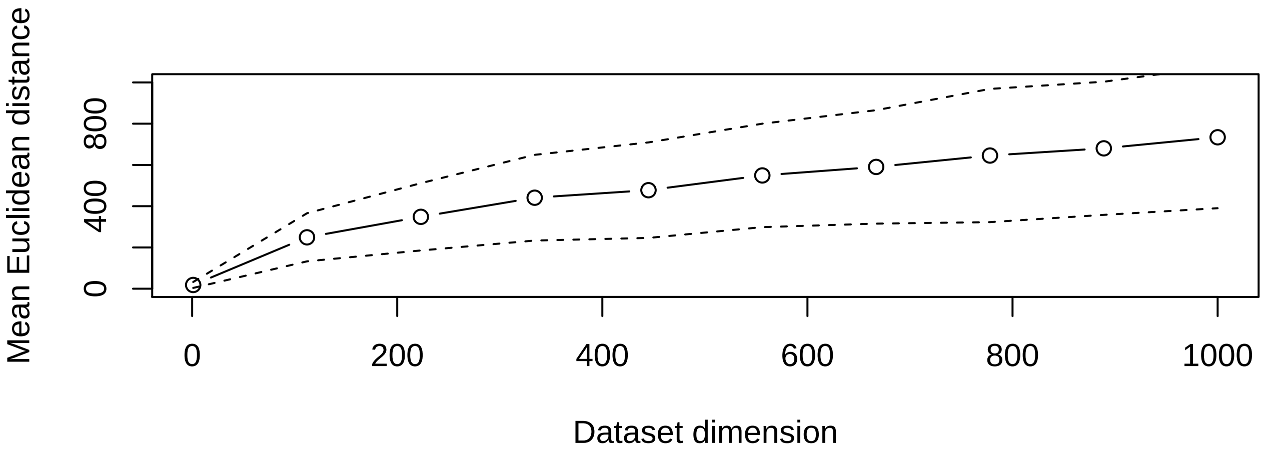

Figure 7: Q3. Variation in mean Euclidean distance between proposal data sets D and an observed data set $\mathcal{D}$. $\mathcal{D}$ was generated using the Poisson distribution with $\theta = \theta_0$. The sample mean at each dimension was computed over $1,000$ data sets D generated using the Poisson parameterised by θ drawn from the prior $\pi(\theta)$. The dashed lines represent the mean ± one sample standard deviation of the observed distances. We can see that an increased threshold will be required at higher dimensions to achieve a given acceptance rate.

```r
# ~ Markov-chain posterior approximation experiment ~ #
set.seed(708432)

# Prior hyperparameters
omega <- 7
k     <- 10

# Utility functions
rprior     <- function(n) rgamma(n, shape=k, scale=omega)
dprior     <- function(t) dgamma(t, shape=k, scale=omega)
dposterior <- function(t) dgamma(x=t, shape=k+sum(x), scale=omega/(n*omega + 1))
pposterior <- function(t) pgamma(q=t, shape=k+sum(x), scale=omega/(n*omega + 1))

# Metropolis-Hastings algorithm
f          <- function(t) log(dprior(t)) + sum(log(dpois(x, lambda=t)))

realizeChain <- function(L, sigma=1) {
                  t    <- rep(NA, L)
                  t[1] <- rprior(1)
                  for(i in 2:L){
                    t_dash <- rnorm(1, mean=t[i-1], sd=sigma)
                    if (log(runif(1)) <= (f(t_dash) - f(t[i-1]))){
                      t[i] <- t_dash # Accept
                    }
                    else  t[i] <- t[i-1] # Reject
                  }
                  return(t)
                }
```

```
M  <- 100 # Number of chain realizations
Nb <- 1e3 # Burn-in period
L  <- 1e4 + Nb # Maximum chain length
Theta <- matrix(rep(NA, M*L), nrow=M) # Chain realizations
for(i in 1:M) Theta[i, ] <- realizeChain(L)

# Compute KS statistics
l          <- seq(10, L-Nb, by=10) # l = 10, 20, ...
KS         <- matrix(rep(NA, M*length(l)), nrow=M)

for (i in 1:M){
  for (j in 1:length(l)){
    Fhat_l   <- ecdf(Theta[i, Nb:(Nb+l[j])])
    KS[i, j] <- max(abs(Fhat_l(theta_grid) - pposterior(theta_grid)))
  }
  print(i)
}
```

```r
# ~ ABC posterior approximation experiment ~ #
#          - Sum, varying threshold -
set.seed(2398743)

Nabc    <- 1e5  # Number of proposals
epsilon <- round(10^(seq(1, log(10^3, base=10), length.out=20)), 0) # Acc. threshold
D_cal   <- x # Caligraphic D
S_star  <- function(x) sum(x) # Sufficient statistic
rho     <- function(a, b) sqrt(sum((a - b)^2)) # L2 distance
accepted_theta <- list() # Samples from ABC posterior

# Compute summary statistic for observed data
S_D <- S_star(D_cal) # 1869

for (j in 1:length(epsilon)){
  acc            <- c()
  for (i in 1:Nabc){
    proposed_theta <- rprior(1) # Sample from prior
    D              <- rpois(n, lambda=proposed_theta) # Simulate data set
    if (rho(S_D, S_star(D)) <= epsilon[j]){ # Compute summary, accept/reject
      acc <- c(acc, proposed_theta)
    }
  }
  accepted_theta[[j]] <- acc
}

# Compute KS statistics
theta_grid <- seq(60, 90, by=0.01)
KS    <- c()
acc_n <- c() # Number of proposals accepted
for (j in 1:length(epsilon)){
  Fhat_l <- ecdf(accepted_theta[[j]])
  KS     <- c(KS, max(abs(Fhat_l(theta_grid) - pposterior(theta_grid))))
  acc_n  <- c(acc_n, length(accepted_theta[[j]]))
}
```

```r
# ~ ABC posterior approximation experiment ~ #
#       - Median, varying threshold -
set.seed(2398743)

Nabc    <- 1e5  # Number of proposals
epsilon <- round(10^(seq(-2, 2, length.out=20)), 3) # Acc. threshold
D_cal   <- x # Caligraphic D
S_med   <- function(x) median(x) # Non-sufficient summary statistic
rho     <- function(a, b) sqrt(sum((a - b)^2)) # L2 distance
accepted_theta <- list() # Samples from ABC posterior

# Compute summary statistic for observed data
S_D <- S_med(D_cal) # 75

for (j in 1:length(epsilon)){
  acc            <- c()
  for (i in 1:Nabc){
    proposed_theta <- rprior(1) # Sample from prior
    D              <- rpois(n, lambda=proposed_theta) # Simulate data set
    if (rho(S_D, S_med(D)) <= epsilon[j]){ # Compute summary, accept/reject
      acc <- c(acc, proposed_theta)
    }
  }
  accepted_theta[[j]] <- acc
}

# Compute KS statistics
theta_grid <- seq(60, 90, by=0.01)
KS    <- c()
acc_n <- c() # Number of proposals accepted
for (j in 1:length(epsilon)){
  Fhat_l <- ecdf(accepted_theta[[j]])
  KS     <- c(KS, max(abs(Fhat_l(theta_grid) - pposterior(theta_grid))))
  acc_n  <- c(acc_n, length(accepted_theta[[j]]))
}
```