Bayesian Posterior Comprehension via Message from Monte Carlo

Leigh J. Fitzgibbon, David L. Dowe and Lloyd Allison
School of Computer Science and Software Engineering
Monash University, Clayton, VIC 3800, Australia
{leighf,dld,lloyd}@bruce.csse.monash.edu.au

Abstract

We discuss the problem of producing an epitome, or brief summary, of a Bayesian posterior distribution - and then investigate a general solution based on the Minimum Message Length (MML) principle. Clearly, the optimal criterion for choosing such an epitome is determined by the epitome's intended use. The interesting general case is where this use is unknown since, in order to be practical, the choice of epitome criterion becomes subjective. We identify a number of desirable properties that an epitome could have - facilitation of point estimation, human comprehension, and fast approximation of posterior expectations. We call these the properties of *Bayesian Posterior Comprehension* and show that the Minimum Message Length principle can be viewed as an epitome criterion that produces epitomes having these properties. We then present and extend Message from Monte Carlo as a means for constructing instantaneous Minimum Message Length codebooks (and epitomes) using Markov Chain Monte Carlo methods. The Message from Monte Carlo methodology is illustrated for binary regression, generalised linear model, and multiple change-point problems.

Keywords: Bayesian, Minimum Message Length, MML, MCMC, RJMCMC, Message from Monte Carlo, MMC, posterior summary, epitome, Bayesian Posterior Comprehension

1 Introduction

The Minimum Message Length (MML) principle [Wallace and Boulton, 1968, 1975, Wallace and Freeman, 1987, Wallace and Dowe, 1999] is often considered to be a Bayesian method for model class selection and (invariant) point estimation. This is apparently due to the method of the widely used MML87 approximation [Wallace and Freeman, 1987]. Such a description is a generalisation that does not hold for all MML approximations since for strict MML [Wallace and Boulton, 1975] [Wallace and Freeman, 1987, page 242] [Wallace and Dowe, 1999], and some other approximations (including those we present in this paper), the notion of model selection does not exist.

A more general description of MML methods is that they give an invariant criterion for selecting a countable set of weighted point estimates from a Bayesian posterior distribution/density. The derivation and definition of the "objective" functions found in the many MML approximations are motivated by ideas from information theory and Bayesian statistics. What all of the MML approximations have in common is that they attempt to estimate the codebook which minimises the expected length of a special[1] two-part message encoding the point estimate and data.

The MML principle complements standard Bayesian methods. It provides an invariant and "objective" means to construct an epitome, or brief summary, of a posterior distribution. Such an epitome can be used for point estimation, human comprehension and for fast approximation of posterior expectations. In this paper we investigate a Markov Chain Monte Carlo-based methodology called Message from Monte Carlo (MMC) [Fitzgibbon, Dowe, and Allison, 2002a,b] that is being developed for constructing MML epitomes. The contribution of this paper is in the refinement of the MMC method - we use more accurate approximations, give extensions to the algorithms, and investigate the behaviour of the method on new problems.

In the first section we briefly define the problem of constructing an epitome of a posterior distribution. We then discuss the use of the MML instantaneous codebook as an epitome that has desirable properties which we describe as Bayesian Posterior Comprehension. In Section 3 we introduce elements of the Message from Monte Carlo (MMC) methodology. In Section 4 we give an MMC algorithm suitable for unimodal likelihood functions of fixed dimension. The algorithm is demonstrated for parameter estimation in a binomial regression problem and link selection in a generalised linear model. Section 5 briefly discusses an algorithm suitable for multimodal likelihood functions of fixed dimension. An algorithm for variable dimension posterior distributions is given in Section 6 and demonstrated using a multiple change-point estimation problem and synthetic data. Further work is discussed in Section 7, and the conclusion can be found in Section 8.

2 Bayesian Posterior Comprehension

Suppose we wish to construct an epitome, or brief summary, of a posterior distribution that could be used as a substitute for the full posterior distribution for all subsequent analyses. A general set of properties that we might reasonably expect from such an epitome are the facilitation of:

1. Point estimation.

2. Human comprehension (i.e., human insight and understanding).

3. Approximation of posterior expectations.

In this paper we will refer to these properties as the properties of Bayesian Posterior Comprehension (BPC). For an epitome with BPC properties to be of any use, it must

[1] The two-part messages are special in that, of the class of two-part messages, there is purposely an inefficiency in the second part of the message. The inefficiency arises because each entry in the codebook can be used to encode any data, not just the data that it is optimal for.

contain as much information about the posterior distribution as possible. We note that we have ruled out choosing the epitome criterion on a case by case basis by requiring that the epitome be suitable for all subsequent analyses. Otherwise we would choose the epitome criterion with minimum expected loss given time, computation and other constraints.

An epitome could take many forms, therefore we must first settle representational issues. Three alternative representations that could be considered are:

1. Approximate the posterior distribution by fitting a parametric distribution to it.

2. Sample from the posterior distribution - the sample is then the epitome.

3. Choose a small weighted subset of the parameter space where the weights somehow represent the goodness of each estimate.

The first representational option could be the most succinct and easily interpreted by an operator for many posterior distributions. However, there would be other more complicated posterior distributions such as that for a non-trivial change-point problem where the epitome would be quite complicated and difficult to comprehend by a human. This would therefore violate the second property of BPC (human comprehension). Facilitation of the third property of BPC may also be difficult.

Representational option number two (sampling) is now a routine part of Bayesian inference due to Markov Chain Monte Carlo methods [Gilks, Richardson, and Spiegelhalter, 1996] but is not as succinct a representation as we would like. This affects the human comprehension property and also the computation time required for approximating posterior expectations.

The third representation is attractive because if the set of estimates and weights are chosen correctly then it can fulfill the requirements of BPC. We will require that the weights assigned to each estimate somehow correspond to the posterior probability associated with the estimate. Therefore we seek a weighted subset of the parameter space:

$$\varepsilon = \{(\theta_1, w_1), ..., (\theta_K, w_K)\} \tag{1}$$

where the θ_i are associated with good posterior probability mass and their weights represent their goodness (a function of their probability mass) as an estimate. Such an epitome can be used for point estimation - if we are interested in inferring the single best model then we can use the θ_i with the greatest weight. If the size of the set is small (i.e., K is small) then it can be used for posterior comprehension, as a human could inspect the set of estimates and their weights to get an understanding, and overview, of the posterior distribution. Posterior expectations could be approximated by normalising the weights and treating the set as a distribution.

Choosing a weighted set of estimates having BPC properties is a multi-objective problem. The size of the set will have a significant impact on how the conflicting BPC objectives are satisfied. If the set is too small then approximated posterior expectations will be poor and human comprehension may also suffer. If it is too large then approximated posterior expectations will require more computation time and human comprehension may again suffer.

One Bayesian approach to BPC where the parameter space is a union of subspaces of differing dimension (variable dimension posterior) would be to return as the epitome, the mode from each subspace with a weight equal to the posterior probability of the subspace. This would be less than ideal when the posterior distribution contains a multimodal subspace, since important parts of the posterior may not be represented in the epitome. Another problem with this approach is that the weights can be misleading since it is possible that a subspace containing a large amount of posterior probability also contains a mode that lies in an area of relatively poor posterior probability mass.

An approach that meets some of the requirements of BPC is Occam's Window (OW) [Madigan and Raftery, 1994, Raftery, Madigan, and Hoeting, 1997]. The Occam's Window algorithm was devised primarily to allow for fast Bayesian Model Averaging. The algorithm is based on selecting a small set of subspaces from the parameter space by using posterior sampling. The strategy is not ideally suited for BPC, since in terms of point estimates, it would suffer from the same problems discussed in the previous paragraph.

[Wallace and Freeman, 1987, Wallace and Dowe, 1999, Wallace and Boulton, 1975, 1968]

The Minimum Message Length (MML) principle, which was briefly discussed in the introduction, can be used for constructing a BPC epitome. MML methods attempt to estimate a codebook - consisting of a countable set of point estimates, θ_i, and their quasi-prior probabilities, p_i. The definition of this set is defined, in information-theoretic terms, as the codebook which minimises the expected length of a special two-part message encoding the point estimate and data [Wallace and Freeman, 1987, Wallace and Dowe, 1999, Wallace and Boulton, 1975, 1968]. We assume that there exists a sender and a receiver that wish to communicate the observed data over a noiseless coding channel and that they share the codebook. Coding theory tells us that an event with probability p can be encoded in a message with length $-\log p$ nits using[2] an ideal Shannon code. So in theory, the sender can transmit some observed data to the receiver in a two part message. In the first part, the sender transmits a codeword corresponding to a point estimate from the codebook. This requires a message of length $-\log p_i$ nits. The sender then transmits the data encoded using the (already) stated estimate in the second part. This requires a message of length $-\log f(x|\theta_i)$ nits, where $f(.|.)$ is the usual statistical likelihood function. Therefore the total message length (MessLen) of the transmission encoding an hypothesis, θ_i, and the data, x, is

$$MessLen(\theta_i, x) = -\log p_i - \log f(x|\theta_i) \tag{2}$$

We expect the sender to transmit the data using the estimate from the codebook that has the minimum message length (i.e. $argmin_{\theta_i} MessLen(\theta_i, x)$).

In order to minimise the length of these two-part messages on average, we seek the codebook that has minimum expected message length. This creates a trade-off between model complexity and goodness of fit. For example, if you increase the number of entries in the codebook then the expected length of the first part of the message increases (but the expected length of the second part encoding the data decreases). If you decrease the number of entries in the codebook then you get the opposite effect[3].

[2] If we use base-2 logarithms then the message length is measured in bits. Throughout the paper we use natural logarithms, so the message length is measured in $\log_e 2$ bits, or nits [Boulton and Wallace, 1970].

[3] Assuming that the codebook entries are optimal for the given codebook size.

To strictly minimise the expected message length one must create a codebook that can be used to encode any data from the dataspace. This is not computationally practical as a general method of inference (see, e.g., [Farr and Wallace, 2002]). In practice, most MML approximations only attempt to estimate the entries of the codebook that are close to the minimum message length. It is this small, *instantaneous*, codebook that corresponds to an epitome that has the BPC properties. The weights in the MML epitome can be calculated by converting from message lengths to (unnormalised) probabilities - i.e., by taking the inverse log (antilog) of the negative message length

$$\varepsilon = \{(\theta_1, antilog(-MessLen(\theta_1, x))), ..., (\theta_K, antilog(-MessLen(\theta_K, x)))\} \tag{3}$$

We note that this MML epitome is a function of the MML codebook and the observed data (entering through the message length). In the following sections we describe how to create such codebooks using a recent methodology called Message from Monte Carlo. We also illustrate the use of the method with a variety of examples so that the reader may get a feel for the use of the MML instantaneous codebook.

3 Message from Monte Carlo

In previous work [Fitzgibbon, Dowe, and Allison, 2002b,a] we have presented the Message from Monte Carlo (MMC) methodology, a general methodology for performing minimum message length inference using posterior sampling and Monte Carlo integration. In this section we describe the basic elements of the Message from Monte Carlo methodology. These elements will be used in the algorithms given in the following sections.

The basic idea is to partition a sample from the posterior distribution of the parameters into uncertainty regions representing entries in the MML instantaneous codebook. Each region has a point estimate which characterizes the models in the region. The point estimate is chosen as the minimum prior expected Kullback-Leibler distance estimate over the region. The regions are chosen so that the models contained within a region are similar in likelihood and Kullback-Leibler distance (to the point estimate).

Each region also has an associated message length which can be considered as the negative logarithm of the weight attributed to the region. The posterior epitome is the set of point estimates (one for each region) weighted by the antilog of the negative message length.

The message length approximation that is generally used in MMC is Dowe's MMLD minimum message length approximation [Fitzgibbon, Dowe, and Allison, 2002a, section 2.4]. Given an uncertainty region of the parameter space, R, prior distribution, $h(\theta)$, and likelihood function, $f(x|\theta)$, the message length of the region, R, is

$$\text{MMLD MessLen } = -\log\left(\int_R h(\theta)\, d\theta\right) - \frac{\int_R h(\theta) \cdot \log f(x|\theta)\, d\theta}{\int_R h(\theta)\, d\theta} \tag{4}$$

for continuous parameter space, or

$$\text{MMLD MessLen } = -\log\left(\sum_{\theta \in R} h(\theta)\right) - \frac{\sum_{\theta \in R} h(\theta) \cdot \log f(x|\theta)}{\sum_{\theta \in R} h(\theta)} \tag{5}$$

for discrete parameter spaces[4].

The MMLD message length equation can be seen to contain two terms. The first term is the negative logarithm of the integral of the prior over the region. It approximates the length of the first part of the MML message (i.e. $-\log p_i$) from Equation 3. The second term is the prior expected negative logarithm of the likelihood function over the region. This approximates the MML message second part and is an average because we do not know the true estimate used in the first part. A shorter message length is to be preferred and involves a trade-off between the first and second terms. Consider a region growing down from the mode in a unimodal likelihood function. As the region grows the first term will decrease (as the integral of the prior increases) but the second term will increase (as the likelihood of the models added to the region decreases). The MMLD message length attempts to take into account the probability mass associated with the region surrounding a point estimate (rather than the point estimate's density for example).

To calculate the MMLD message length we use importance sampling and Monte Carlo integration. We sample from the posterior distribution of the parameters

$$S = \{\theta_t : t = 1, ..., N\} \tag{6}$$

and then choose a subset of this sample, Q, to implicitly define the uncertainty region, R. The first part of the message length can then be approximated by

$$\text{MMC 1st Part } \approx -\log\left(\frac{\frac{1}{N}\sum_{\theta\in Q} h(\theta)I(\theta)^{-1}}{\frac{1}{N}\sum_{\theta\in S} h(\theta)I(\theta)^{-1}}\right) = -\log\left(\frac{\sum_{\theta\in Q} f(x|\theta)^{-1}}{\sum_{\theta\in S} f(x|\theta)^{-1}}\right) \tag{7}$$

where $I(.)$ is the importance sampling distribution (here we use the posterior, $I(\theta) \propto h(\theta)f(x|\theta)$). This estimate does not require that the prior, $h(.)$, or the importance sampling distribution be normalised.

The second part is approximated using

$$\text{MMC 2nd Part } \approx -\frac{\sum_{\theta\in Q} h(\theta)I(\theta)^{-1}\log f(x|\theta)}{\sum_{\theta\in Q} h(\theta)I(\theta)^{-1}} = -\frac{\sum_{\theta\in Q} f(x|\theta)^{-1}\log f(x|\theta)}{\sum_{\theta\in Q} f(x|\theta)^{-1}} \tag{8}$$

These estimates allow the message length to be approximated for some Q. We now discuss how to select the Q that minimises the message length. We first note that if we attempt to minimise Equation 4 we get the following boundary rule

$$-\log f(x|\theta)\Big|_{\theta\in\partial R} = -\frac{\int_R h(\theta)\log f(x|\theta)\,d\theta}{\int_R h(\theta)\,d\theta} + 1 \tag{9}$$

where the boundary, ∂R, of R, is an iso-likelihood contour of f. In other words, the values of $f(x|\theta)$ and of $\log f(x|\theta)$ are constant on ∂R. This boundary rule states that for the region which minimises the message length, the negative log-likelihood at the boundary of R is equal to the prior expected negative log-likelihood over the region plus one. The right hand side can be approximated using Equation 8.

[4] We could also have parameter spaces with both continuous and discrete parameters, and of varying dimension.

For the discrete version of MMLD (Equation 5) the boundary rule (Equation 9) is only an approximation and has the following error

$$err(\theta') = 1 - \frac{\sum_{\theta \in R} h(\theta) + h(\theta')}{h(\theta')} \log \frac{\sum_{\theta \in R} h(\theta) + h(\theta')}{\sum_{\theta \in R} h(\theta)} \tag{10}$$

which involves only the prior.

Since the posterior sampling process has discretised the space we will need to include the *err* term for both discrete and continuous parameter spaces. We can estimate the *err* term using

$$err(\theta') \approx 1 - \frac{\sum_{\theta \in Q} f(x|\theta)^{-1} + f(x|\theta')^{-1}}{f(x|\theta')^{-1}} \log \left(\frac{\sum_{\theta \in Q} f(x|\theta)^{-1} + f(x|\theta')^{-1}}{\sum_{\theta \in Q} f(x|\theta)^{-1}} \right) \tag{11}$$

Due to the use of importance sampling the prior terms have cancelled and the estimate does not directly involve the prior. Intuitively we see that *err* is largest when Q is small and therefore *err* will have the largest effect when the region is initially being formed.

Selection of the optimal region is simple in the unimodal likelihood case since we can order the sample in descending order of likelihood, then start at the element with maximum likelihood and continue to grow the region accepting models into the region that pass the boundary rule test

$$\theta \in Q \text{ iff } -\log f(x|\theta) \leq -\frac{\sum_{\theta \in Q} \log f(x|\theta) f(x|\theta)^{-1}}{\sum_{\theta \in Q} f(x|\theta)^{-1}} + 1 - err(\theta) \tag{12}$$

Such an algorithm is given in the next section. This idea can be similarly extended to the multimodal likelihood case by using order statistics to restrict the regions to be simply connected (briefly discussed in Section 5). For variable dimension posteriors we need a different strategy. We must ensure that regions contain models that are close to the point estimate in Kullback-Leibler distance. This is discussed, and an algorithm given, in Section 6. Now we briefly discuss how to choose the point estimate for a region.

3.1 The Point Estimate

Once the region that minimises the message length is found we need to find a point estimate for the region. Staying true to the compact coding approach we use the minimum prior expected Kullback-Leibler distance estimate since this corresponds to the MMLA estimate (see [Fitzgibbon, Dowe, and Allison, 2002a][section 2.4.1]). This estimate represents a compromise between all of the models in the region and can be considered to be a characteristic model that summarises the region. The prior expected Kullback-Leibler (EKL) distance is

$$\begin{aligned} E_{h(\theta)} \left[KL(\theta, \hat{\theta}) \right] &= \int_R h(\theta) KL(\theta, \hat{\theta}) \, d\theta & (13) \\ &\approx \frac{\sum_{\theta \in Q} f(x|\theta)^{-1} KL(\theta, \hat{\theta})}{\sum_{\theta \in Q} f(x|\theta)^{-1}} & (14) \end{aligned}$$

The MML estimate is the $\hat{\theta}$ that gives the minimum EKL distance

$$\hat{\theta} = argmin_{\hat{\theta}\in R} E_{h(\theta)} \left[KL(\theta, \hat{\theta}) \right] \tag{15}$$

This estimate could be found by simulation (see, e.g. [Dowe, Baxter, Oliver, and Wallace, 1998] - note that they use the posterior expectation). It could also be found directly using Newton-Raphson type algorithms (as used in the next section). Both of these require that we know the parametric form of the estimate - making them less desireable for use with variable dimension posteriors. An alternative that does not have such a requirement is to find the element of Q that has the minimum EKL distance. This algorithm was used in the MMC algorithm from [Fitzgibbon, Dowe, and Allison, 2002a][section 3.4]. An exhaustive search (i.e. $argmin_{\hat{\theta}\in Q} E_{h(\theta)} \left[KL(\theta, \hat{\theta}) \right]$), using Equation 14, requires quadratic time in $|Q|$ although simple strategies can be employed to reduce this considerably.

In practice it is often beneficial to use the posterior expected Kullback-Leibler distance

$$\begin{aligned} E_{h(\theta)f(x|\theta)} \left[KL(\theta, \hat{\theta}) \right] &= \frac{\int_R h(\theta) f(x|\theta) KL(\theta, \hat{\theta})\, d\theta}{\int_R h(\theta) f(x|\theta)\, d\theta} & (16) \\ &\approx \frac{1}{|Q|} \sum_{\theta \in Q} KL(\theta, \hat{\theta}) & (17) \end{aligned}$$

since the Monte Carlo estimate is better behaved.

4 Unimodal Likelihood Function

This section describes an MMC algorithm suitable for problems where the likelihood function is unimodal and of fixed dimension. A simple algorithm for finding the region with minimum message length is described, along with a Newton-Raphson algorithm for finding the point estimate.

For the unimodal likelihood function case the minimising MMLD region can be found using Algorithm 1. This algorithm is based on Algorithm 1 from [Fitzgibbon, Dowe, and Allison, 2002a, page 12] but has been modified to use the more accurate message length approximation described in the previous section. The algorithm is fast and efficient requiring a single pass through the sample. It produces a single region.

Choosing the point estimate can be more difficult than finding the region. For continuous parameter spaces of fixed dimension the Newton-Raphson algorithm is suitable. The Newton-Raphson method requires an initial estimate, $\theta^{(0)}$, for which we can use the element of the sample with maximum likelihood. Based on Equation 14, and using the notation $\hat{\theta} = (\hat{\vartheta}_1, ..., \hat{\vartheta}_d)$, each iteration we update $\hat{\theta}^{(k+1)} = \hat{\theta}^{(k)} + d\hat{\theta}^{(k)}$ by solving the following linear system for $d\hat{\theta}^{(k)}$:

$$J \times d\hat{\theta}^{(k)} = -r \tag{18}$$

Algorithm 1. *Pseudo-code for optimising the MMLD message length in the unimodal likelihood case. The algorithm is given as pseudo-code only and in practice extra care would be taken in its implementation to avoid numerical problems.*

```
sample from the posterior distribution $S = \{\theta_t : t = 1, ..., N\}$
sort the sample such that $f(x|\theta_t) \geq f(x|\theta_{t+1})$
1st numer $\leftarrow f(x|\theta_0)^{-1}$
1st denom $\leftarrow \sum_{\theta \in S} f(x|\theta)^{-1}$
2nd numer $\leftarrow -f(x|\theta_0)^{-1} \times \log f(x|\theta_0)$
2nd denom $\leftarrow f(x|\theta_0)^{-1}$
2nd length $\leftarrow$ 2nd numer / 2nd denom
$Q \leftarrow \{\theta_0\}$
$i \leftarrow 1$
while $(i < N)$
    corr $\leftarrow \frac{\text{1st numer}+f(x|\theta_i)^{-1}}{f(x|\theta_i)^{-1}} \log\left(\frac{\text{1st numer}+f(x|\theta_i)^{-1}}{\text{1st numer}}\right)$
    if $(-\log f(x|\theta_i) \leq \text{2nd length} + \text{corr})$ then
        1st numer $\leftarrow$ 1st numer $+f(x|\theta_i)^{-1}$
        2nd numer $\leftarrow$ 2nd numer $-f(x|\theta_i)^{-1} \times \log f(x|\theta_i)$
        2nd denom $\leftarrow$ 2nd denom $+ f(x|\theta_i)^{-1}$
        2nd length $\leftarrow$ 2nd numer / 2nd denom
        $Q \leftarrow Q \cup \{\theta_i\}$
    end
    $i \leftarrow i + 1$
end
```

end

$$J = \begin{bmatrix} \sum_{\theta \in Q} f(x|\theta)^{-1} \left(\frac{\partial^2 KL(\theta,\hat{\theta})}{\partial\hat{\vartheta}_1 \partial\hat{\vartheta}_1} \Big|_{\hat{\theta}^{(k)}} \right) & \cdots & \sum_{\theta \in Q} f(x|\theta)^{-1} \left(\frac{\partial^2 KL(\theta,\hat{\theta})}{\partial\hat{\vartheta}_1 \partial\hat{\vartheta}_d} \Big|_{\hat{\theta}^{(k)}} \right) \\ \vdots & \ddots & \vdots \\ \sum_{\theta \in Q} f(x|\theta)^{-1} \left(\frac{\partial^2 KL(\theta,\hat{\theta})}{\partial\hat{\vartheta}_d \partial\hat{\vartheta}_1} \Big|_{\hat{\theta}^{(k)}} \right) & \cdots & \sum_{\theta \in Q} f(x|\theta)^{-1} \left(\frac{\partial^2 KL(\theta,\hat{\theta})}{\partial\hat{\vartheta}_d \partial\hat{\vartheta}_d} \Big|_{\hat{\theta}^{(k)}} \right) \end{bmatrix} \tag{19}$$

$$d\hat{\theta}^{(k)} = \begin{bmatrix} d\hat{\vartheta}_0^{(k)} \\ \vdots \\ d\hat{\vartheta}_d^{(k)} \end{bmatrix}, \ r = \begin{bmatrix} \sum_{\theta \in Q} f(x|\theta)^{-1} \left(\frac{\partial KL(\theta,\hat{\theta})}{\partial\hat{\vartheta}_1} \Big|_{\hat{\theta}^{(k)}} \right) \\ \vdots \\ \sum_{\theta \in Q} f(x|\theta)^{-1} \left(\frac{\partial KL(\theta,\hat{\theta})}{\partial\hat{\vartheta}_d} \Big|_{\hat{\theta}^{(k)}} \right) \end{bmatrix} \tag{20}$$

In the following "Dog Shock Experiment" example we have used a numerical approximation for the second derivatives in J. In this example we also work in the canonical exponential form. The canonical exponential family (see [Bernardo and Smith, 1994]) of distributions have the following form

$$p(y|\psi) = a(y)e^{\psi \bullet y - b(\psi)} \text{ where } b(\psi) = \ln \int_y e^{\psi \bullet y} a(y) dy \tag{21}$$

Many commonly used distributions can be expressed as members of the exponential family and the Kullback-Leibler distance simplifies to

$$\begin{aligned} KL(\psi, \hat{\psi}) &= -H(\psi) - E_{\psi}\left[\log a(y)\right] + b(\hat{\psi}) - \hat{\psi} \bullet E_{\psi} y \\ &= -b(\psi) + b(\hat{\psi}) + (\psi - \hat{\psi}) \bullet E_{\psi} y \\ &= -b(\psi) + b(\hat{\psi}) + (\psi - \hat{\psi}) \bullet \nabla b(\psi) \qquad (22) \end{aligned}$$

We now illustrate the use of the unimodal MMC algorithm on two simple problems taken from the Bayesian Inference Using Gibbs Sampling (BUGS) [Gilks, Thomas, and Spiegelhalter, 1994] examples. The first is a parameter estimation problem involving a generalised linear model for binary data. The second is a parameter estimation and model selection problem involving a generalised linear model with three plausible link functions. WinBugs13 is used to do the sampling in both examples with a 1000 update burn-in and a final sample size of 10000.

4.1 Example: Dog Shock Experiment

In this example we apply the unimodal MMC algorithm to the dog shock learning model from Lindsey [1994]. While this example is quite trivial, it is intended to illustrate how the unimodal MMC algorithm works. The sampler for this example can be found as BUGS example "Dogs: loglinear model for binary data". Lindsey [1994] analysed the data from the Solomon-Wynne experiment on dogs. The experiment involved placing a dog in a box with a floor through which a non-lethal shock can be applied. The lights are turned out and a barrier raised. The dog has 10 seconds to jump the barrier and escape otherwise it will be shocked due to a voltage being applied to the floor. The data consists of 25 trials for 30 dogs. A learning model is fitted to the data where the probability that the i^{th} dog receives a shock π_k at trial k is based on the number of times it has previously avoided being shocked x_{ik} and the number of previous shocks $k - x_{ik}$ by the model

$$\pi_{ik} = \kappa^{x_{ik}} \times \upsilon^{k - x_{ik}} \qquad (23)$$

or equivalently

$$\log(\pi_{ik}) = \alpha x_{ik} + \beta(k - x_{ik}) \qquad (24)$$

with $\alpha = \log(\kappa)$ and $\beta = \log(\upsilon)$. The important aspect of this model is that $\pi_k = \pi_{k-1}\kappa$ if the shock was avoided at trial $k - 1$, or $\pi_k = \pi_{k-1}\upsilon$ if the shock was received at $k - 1$. In other words the probability of a dog being shocked in the future changes by a factor of κ each time a shock is avoided and by a factor of υ each time a shock occurs.

The unimodal MMC algorithm was run on the output from the BUGS program. The contents of the sample and the optimal region are shown in Figure 1. The message length of the region is 276.49 nits. The region contains 82 percent of the posterior probability. For this simple problem the message length and shape of the region is purely academic since there are no issues of model selection. We are more interested in the point estimate for the region. The following quantities are required for the Newton-Raphson point estimate algorithm (using notation from Equation 21)

$$
\begin{aligned}
\theta &= (\alpha, \beta) && (25)\\
\psi(\theta, x_{ik}) &= \log \frac{\pi_{ik}}{1-\pi_{ik}} && (26)\\
a(y) &= 1 && (27)\\
b(\psi(\theta, x_{ik})) &= \log(1 + e^{\pi}_{ik}) && (28)\\
\frac{\partial b(\psi(\theta, x_{ik}))}{\partial \psi} &= \frac{e^{\psi(\theta, x_{ik})}}{1 + e^{\psi(\theta, x_{ik})}} && (29)\\
\frac{\partial \psi(\theta, x_{ik})}{\partial \alpha} &= x_{ik}(1 - \pi_{ik}) && (30)\\
\frac{\partial \psi(\theta, x_{ik})}{\partial \beta} &= (k - x_{ik})(1 - \pi_{ik}) && (31)\\
\frac{\partial b(\psi(\theta, x_{ik}))}{\partial \alpha} &= \frac{e^{\psi(\theta, x_{ik})} x_{ik}(1 - \pi_{ik})}{1 + e^{\psi(\theta, x_{ik})}} && (32)\\
\frac{\partial b(\psi(\theta, x_{ik}))}{\partial \beta} &= \frac{e^{\psi(\theta, x_{ik})} (j - x_{ik})(1 - \pi_{ik})}{1 + e^{\psi(\theta, x_{ik})}} && (33)\\
& && (34)
\end{aligned}
$$

The Newton-Raphson algorithm converged after six iterations to the estimates $\kappa = 0.788$ and $\upsilon = 0.924$. The estimate for υ corresponds with the posterior mean reported by BUGS and the maximum likelihood estimate from Lindsey [1994] to three decimal places. The estimate for κ differs only in the third decimal place and lies above the mean and below the maximum likelihood estimate as can be seen in Figure 1.

The epitome for this example contains only a single entry with weight 1

$$\varepsilon = \{((\kappa = 0.788, \upsilon = 0.924), 1)\} \tag{35}$$

4.2 Example: Beetle Mortality Data

In this example we use the unimodal MMC algorithm to perform model selection for the BUGS example "Beetles: logistic, probit and extreme value (log-log) model comparison". The example is based on an analysis by Dobson [1983] of binary dose-response data. In an experiment, beetles are exposed to carbon disulphide at eight different concentrations (x_i) and the number of beetles killed after 5 hours exposure is recorded.

Three different link functions for the proportion killed, π_i, at concentration x_i are entertained

$$
\begin{aligned}
\pi_i &= \frac{e^{\alpha + \beta x_i}}{1 + \alpha + \beta x_i} && \text{Logit} && (36)\\
&= \Phi(\alpha + \beta x_i) && \text{Probit} && (37)\\
&= 1 - e^{-e^{\alpha + \beta x_i}} && \text{CLogLog} && (38)
\end{aligned}
$$

Dobson [1983] used the log-likelihood ratio statistic to assess the three link functions for goodness of fit. The test showed that the extreme value log-log link function provided

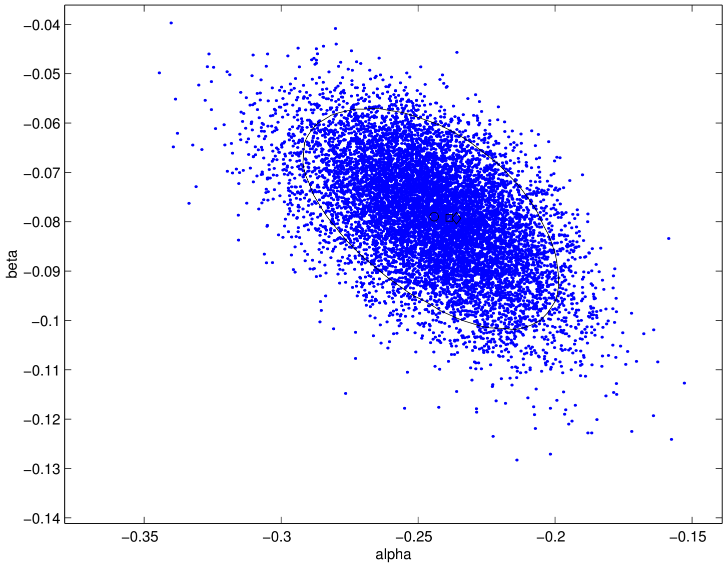


Figure 1: Plot of the posterior sample (10,000 elements) for the dog shock data. The ellipse indicates the estimated MMLD boundary, ∂R, of the region, R. Parameter estimates: posterior mean=circle; maximum likelihood=diamond; MMC (minimum prior expected Kullback-Leibler distance)=square.

the best fit to the data. We ran the unimodal MMC algorithm on 10000 elements sampled from the posterior for each link function. The message lengths and corresponding normalised weights for each link function are given in Table 1. We see that MMC also provides strong support for the extreme value log-log link function, giving it a weight of 0.9. MMC gives more than twice the weight to the probit link compared to the logit link function. This is in contrast to the log-likelihood ratio statistic that gives only slightly more support for the probit model. From the table we can deduce that the peak of the probit model likelihood function contains more probability mass that that of the logit model. In other words the logit model gets less weight by MMC because the parameters lie in a region with slightly less posterior probability mass than the probit.

The epitome for this example contains entries corresponding to each link function

$$\varepsilon = \{(\hat{\theta}_{logit}, 0.03), (\hat{\theta}_{probit}, 0.07), (\hat{\theta}_{cloglog}, 0.9)\} \tag{39}$$

Table 1: MMC analysis of beetle mortality data

Link	**1st part length**	**2nd part length**	**Message length**	**Weight**
Logit	2.098 nits	186.857 nits	188.956 nits	0.03
Probit	1.729 nits	186.219 nits	187.949 nits	0.07
CLogLog	2.145 nits	183.310 nits	185.456 nits	0.90

5 Multimodal Likelihood Function

For multimodal likelihood functions of fixed dimension the unimodal algorithm could be extended to build only contiguous (simply connected) regions using order statistics.

We have not pursued this as we have devised the variable dimension posterior solution, which is more general (see next section). However, a multimodal solution based on order statistics could be investigated in the future.

6 Variable Dimension Posterior

This section describes an algorithm suitable for variable dimension posterior distributions. Unlike the simple unimodal algorithm we must incorporate a Kullback-Leibler distance acceptance test, so that regions contain only models that are similar. Such a constraint follows from Wallace's MMLA approximation (see, e.g. [Fitzgibbon, Dowe, and Allison, 2002a, section 2.2]). This also ensures that the MMC instantaneous codebook corresponds to an epitome with BPC properties. We therefore augment the likelihood-based acceptance rule (Equation 12) to include the following requirement

$$\theta \in Q \text{ iff } KL(\theta, \hat{\theta}) \leq \frac{\sum_{\theta' \in Q} f(x|\theta')^{-1} KL(\theta', \hat{\theta})}{\sum_{\theta' \in Q} f(x|\theta')^{-1}} + 1 - err(\theta) \tag{40}$$

where $err(.)$ is the same $err(.)$ defined in Equation 11, and $KL(.|.)$ is the Kullback-Leibler distance.

In previous work the basic unimodal algorithm was modified to include this Kullback-Leibler distance acceptance rule and to make multiple passes through the sample [Fitzgibbon, Dowe, and Allison, 2002a, page 14]. While this algorithm was found to be satisfactory for the univariate polynomial problem it was applied to, we found that the regions refused to grow for the change-point problem that we consider later in this section. The problem was due to the discrete parameter space - the Kullback-Leibler acceptance rule stopped the regions from growing.

Therefore we have devised a slightly different algorithm that does not suffer from this problem. The algorithm consists of two phases. In the first phase we apply the unimodal algorithm to the sample recursively. That is, we start the unimodal algorithm and keep track of which elements of the sample have been allocated. Once the first region has been formed we store the results and restart the algorithm on the unallocated elements of the sample. This is repeated until all elements of the sample have been allocated to a region. We therefore end up with a set of regions: $U = \{Q_1, ..., Q_K\}$. Since these regions have been formed using the unimodal algorithm some regions will not be pure, they will contain models that are dissimilar (i.e., they will violate Equation 40). We therefore enter phase two of the algorithm where we recursively estimate the point estimate for each region and reassign elements between regions. The recursion stops when no reassignments were made in the last iteration. The reassignment between regions is based on Kullback-Leibler distance. For each element of each region we test whether there is a region whose point estimate is closer in Kullback-Leibler distance. If there is and the element passes the Kullback-Leibler distance acceptance rule (Equation 40) for the candidate region then the element is moved to the candidate region. Phase two of the algorithm is given as pseudo-code in Figure 2. After phase two of the algorithm has completed we are left with an instantaneous MML codebook which defines an epitome having BPC properties.

We now illustrate the use of the algorithm for a multiple change-point problem.

Algorithm 2. *Pseudo-code for optimising the MMLD message length in the variable dimension case: Second phase.*

```
changed ← true
while changed
    changed ← false
    for each $Q$ in $U$ do
        find the point estimate for $Q$ and store it in $\hat{Q}$
    end
    for each $Q$ in $U$ do
        for each $\theta$ in $Q$ do
            for each $Q'$ in $U - \{Q\}$ do
                if $KL(\theta, \hat{Q}') < KL(\theta, \hat{Q})$ and
                $-\log f(x|\theta) \leq -\frac{\sum_{\theta' \in Q'} \log f(x|\theta') f(x|\theta')^{-1}}{\sum_{\theta' \in Q'} f(x|\theta')^{-1}} + 1 - err(\theta)$ and
                $KL(\theta, \hat{Q}') \leq \frac{\sum_{\theta' \in Q'} KL(\theta', \hat{Q}') f(x|\theta')^{-1}}{\sum_{\theta' \in Q'} f(x|\theta')^{-1}} + 1 - err(\theta)$ then
                    move $\theta$ from $Q$ to $Q'$
                    changed ← true
                end
            end
        end
    end
end
```

end

6.1 Example: Multiple Change-Point Model

We now apply a multiple change-point model to synthetic data. In order to apply the MMC algorithm we require a sample from the posterior distribution of the parameters and a function for evaluation of the Kullback-Leibler distance between any two models. The sampler that we use is a Reversible Jump Markov Chain Monte Carlo sampler [Green, 1995] that was devised for sampling piecewise polynomial models by Denison, Mallick, and Smith [1998]. The sampler is simple, fast and relatively easy to implement. The sampler can make one of three possible transitions each iteration:

- Add a change-point
- Remove a change-point
- Move an existing change-point

We fit constants in each segment and use a Gaussian distribution to model the noise. However, rather than include the Gaussian parameters in the sampler we use the maximum likelihood estimates. This means that the only parameters to be simulated are the number of change-points and their locations. Use of the maximum likelihood estimates required us to use a Poisson prior over the number of change-points with $\lambda = 0.5$.

The Kullback-Leibler distance is easily calculated for the piecewise constant change-point model and has time complexity that is linear in the number of change-points.

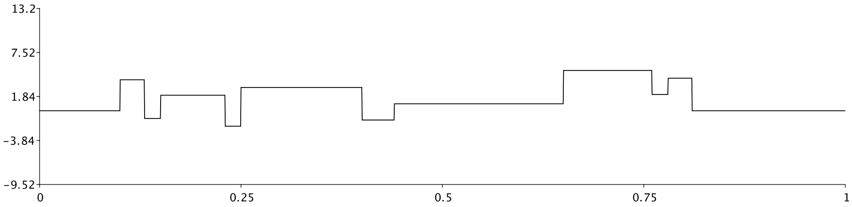


Figure 2: Blocks test function.

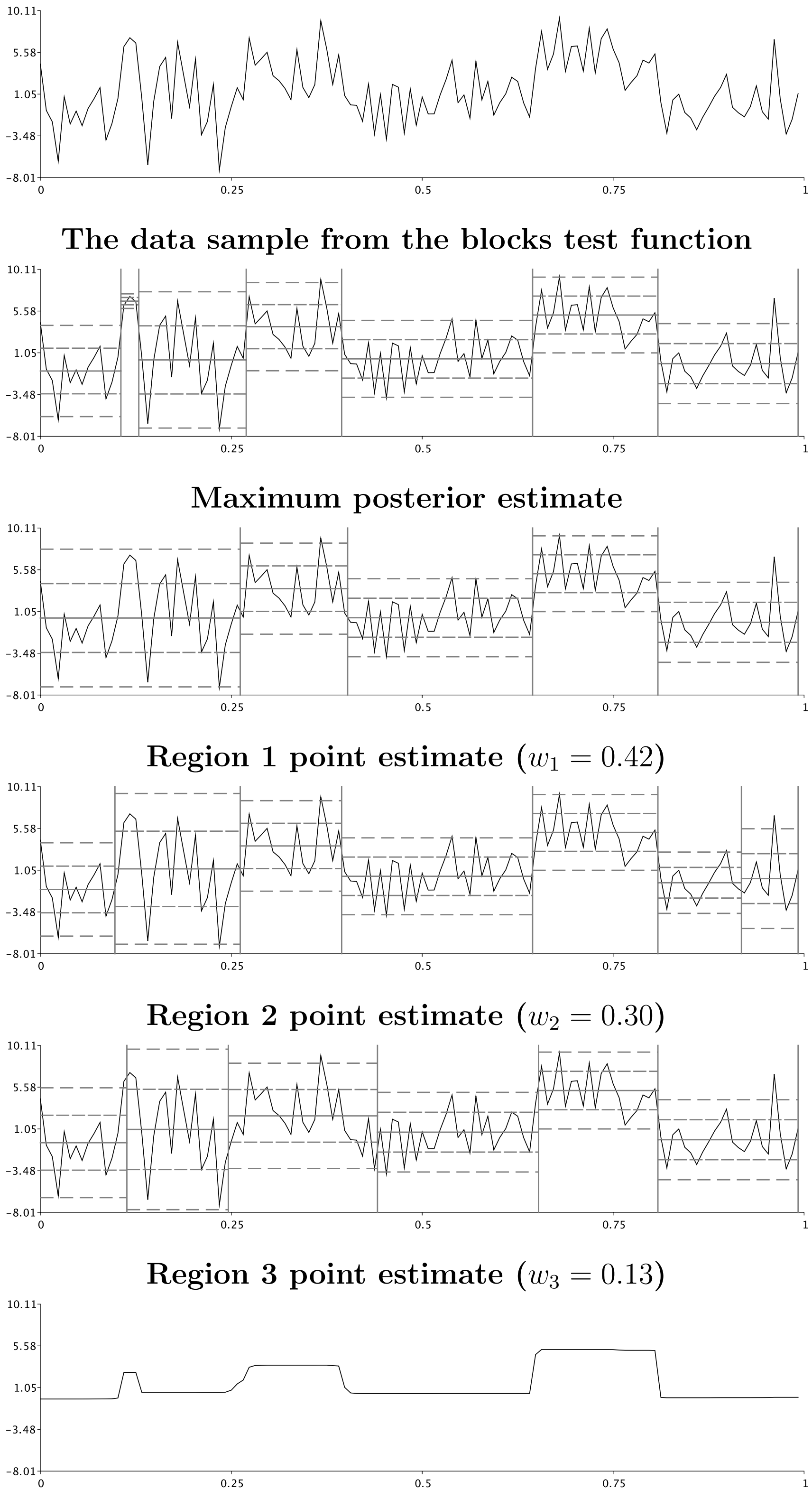
The function that we have used in the evaluation is the "blocks" function from [Donoho and Johnstone, 1994]. The function is illustrated in Figure 2 and consists of eleven change-points over the domain $[0, 1]$. We added Gaussian noise to the blocks function with $\sigma = 2.5$ for all segments. Experiments were conducted for two data sample sizes: $N = 128$ (small) and $N = 2048$ (large). For each of these data samples we simulated 500,000 change-point models after an initial burn-in period of 10000. Every one-hundredth element of the sample was kept, thus reducing the usable sample size to 5000. We then applied the MMC algorithm to each sample.

The main results for the small ($N = 128$) data sample size experiment can be seen in Figure 3. This figure shows the data sample from the blocks function with $\sigma = 2.5$ and $N = 128$. The element of the sample with maximum posterior probability (maximum posterior estimate) is also shown along with the three regions with the greatest weight, and the posterior mean estimate of the function. The epitome contained nine regions. The remaining regions (4-9) are shown in Figure 4. In the figures a change-point is marked using a vertical bar, and for each segment the mean and one and two standard deviation bars are shown allowing a change in the mean or standard deviation estimates for a segment to be easily seen.

With such little data we do not expect the true blocks function to be accurately estimated. The point estimates for regions 1-9 are all reasonable models of the data and represent a good degree of variety. The maximum posterior model closely fits the data in the second segment and would be expected to have a very large Kullback-Leibler distance from the true model. We see that none of the point estimates in the MMC epitome make this mistake.

The point estimates for regions 1 and 7 contain the same number of change-points, yet region 1 is given 42 times the weight of region 7 ($w_1 = 0.42$ and $w_7 = 0.01$). The main difference between the two is in the location of the first two change-points. This illustrates the need for methods to be able to handle multimodal posterior distributions and likelihood functions as this detail would be lost by simply looking at the modal model for each model order. This also occurs for regions 3, 5, 6 and 8, which all contain five change-points.

The main results for the large ($N = 2048$) data sample size experiment can be seen in Figure 5. In this MMC epitome there were 13 regions. Regions 4-14 are shown in Figure 6. In these results we see that the maximum posterior estimate is quite reasonable but lacks one of the change-points that exists in the true function. The point estimate for region 1 is able to detect this change-point and looks almost identical to the true function. The



The data sample from the blocks test function

Maximum posterior estimate

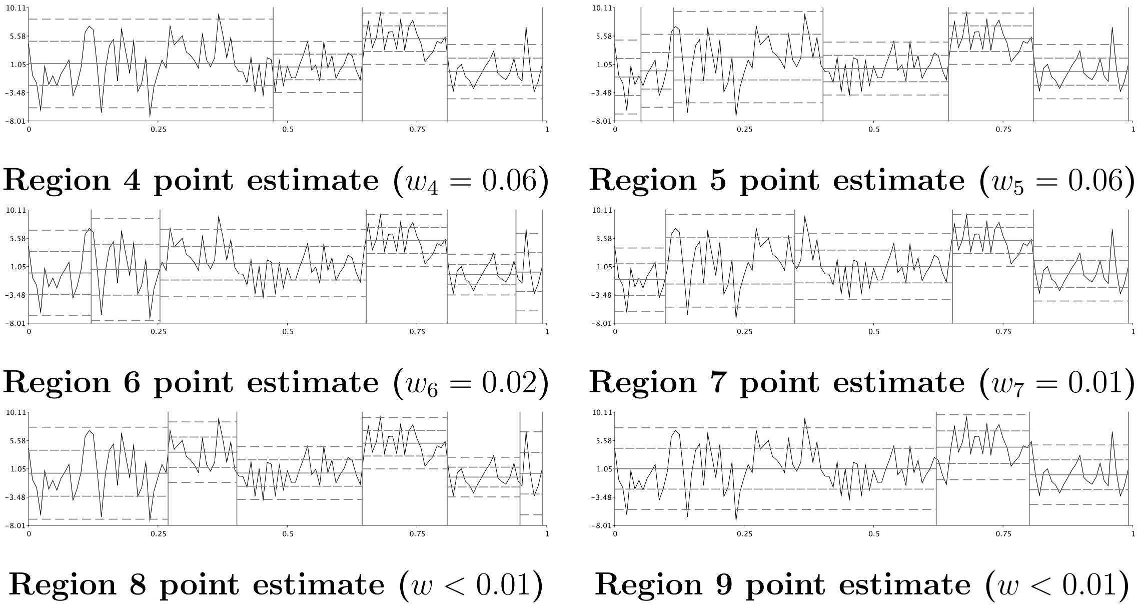
Region 1 point estimate ($w_1 = 0.42$)

Region 2 point estimate ($w_2 = 0.30$)

Region 3 point estimate ($w_3 = 0.13$)

Posterior mean (point-wise) estimate.

Figure 3: Main results for the "blocks" test function with $N = 128$ and $\sigma = 2.5$.



Region 4 point estimate ($w_4 = 0.06$) **Region 5 point estimate ($w_5 = 0.06$)**

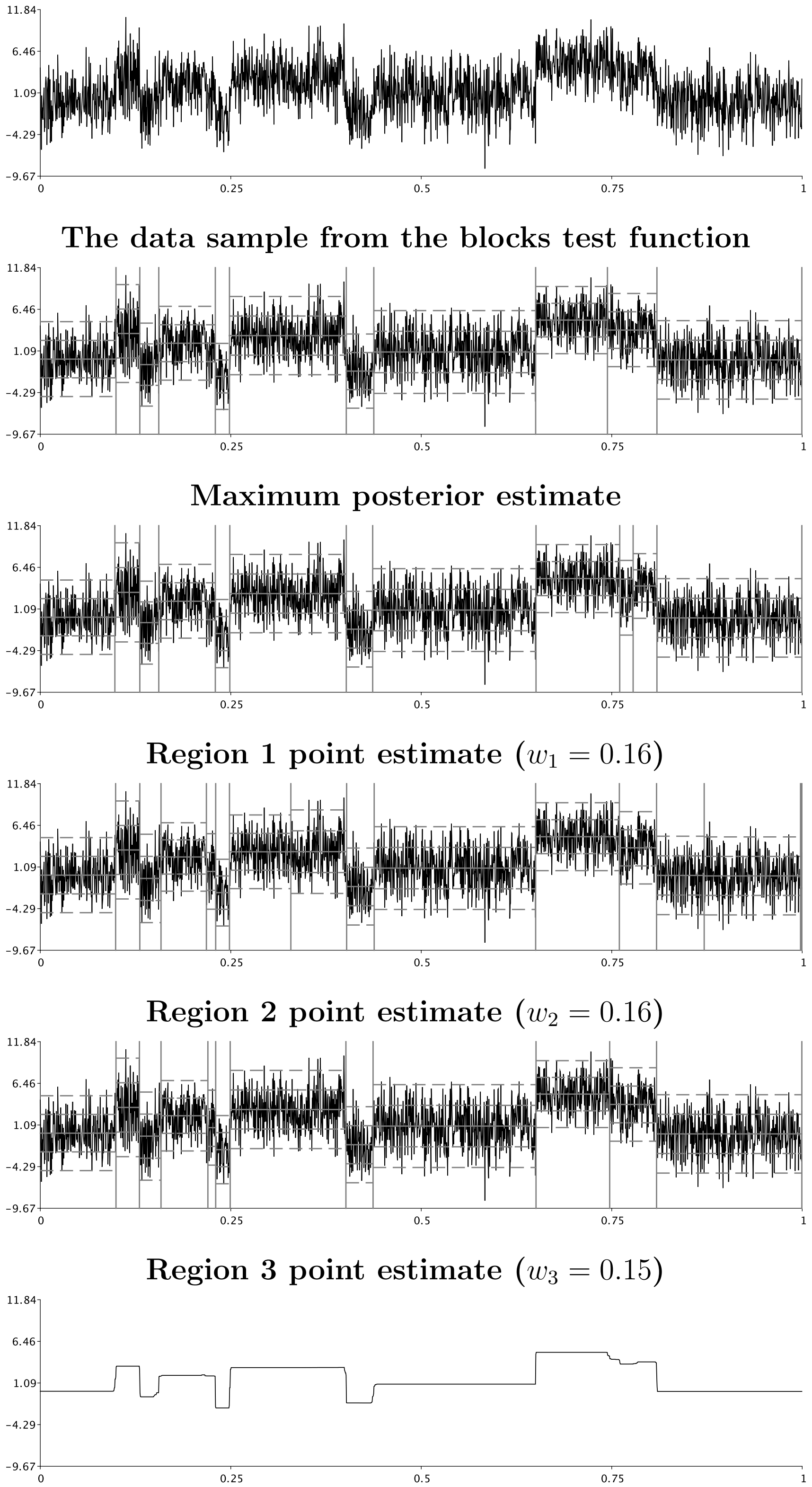
Region 6 point estimate ($w_6 = 0.02$) **Region 7 point estimate ($w_7 = 0.01$)**

Region 8 point estimate ($w < 0.01$) **Region 9 point estimate ($w < 0.01$)**

Figure 4: Regions 4-9 for $N = 128$ and $\sigma = 2.5$.

point estimates for the other regions (2-13) look reasonable and tend to increase in detail. Some of them contain superfluous change-points. This does not damage their predictive ability or Kullback-Leibler distance to the true model, but can be distracting for human comprehension. This problem is discussed further in Section 7.2.

For these examples we have found that the MMC algorithm can produce reasonable epitomes of a variable dimension posterior distribution. For both examples, $N = 128$ and $N = 2048$, we found that the point estimate for the region having the greatest weight in the epitome was closer to the true function than the maximum posterior estimate. We also find that the set of weighted point estimates provides some insight into the sample from the posterior distribution of the parameters and ultimately into the posterior distribution itself. We would also expect that using the set for approximating posterior expectations would be highly accurate.



The data sample from the blocks test function

Maximum posterior estimate

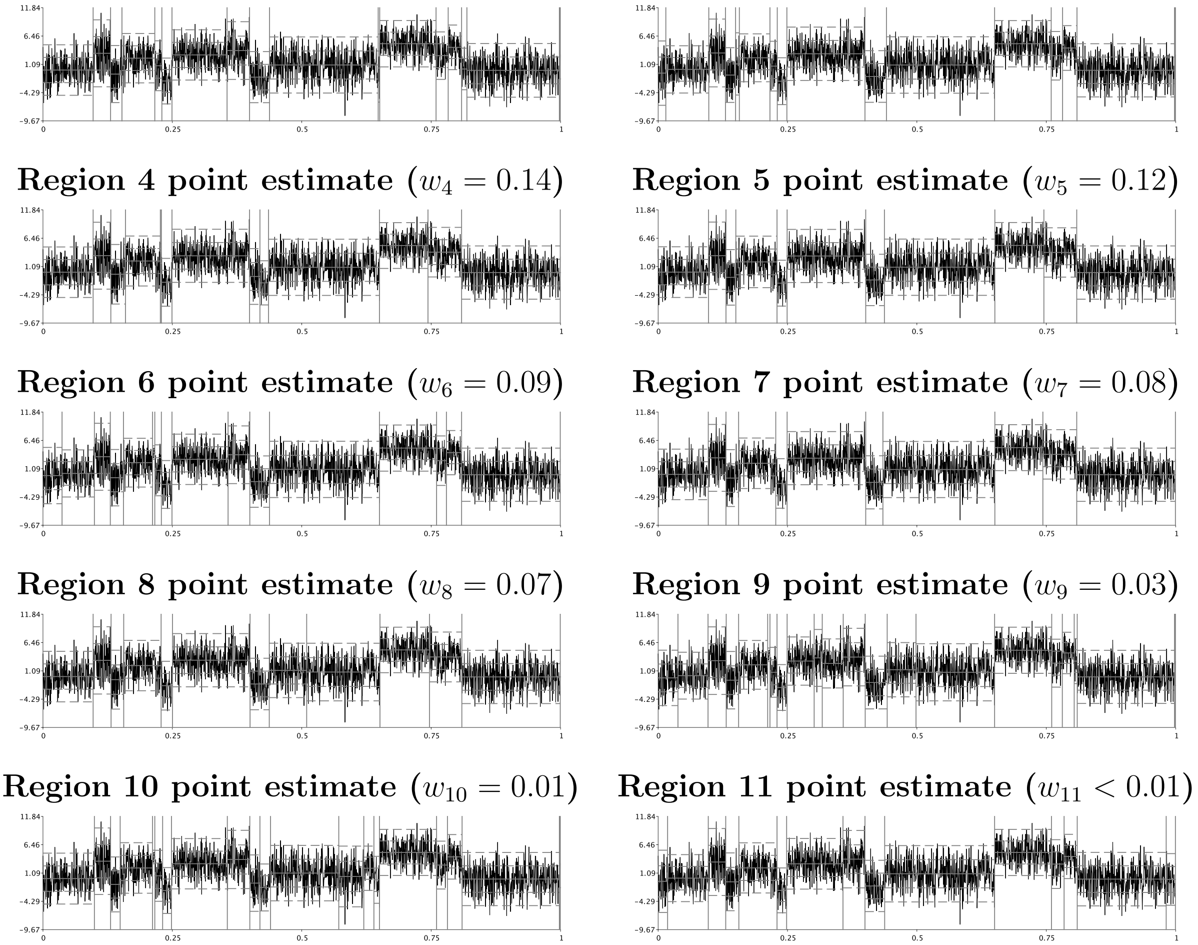
Region 1 point estimate ($w_1 = 0.16$)

Region 2 point estimate ($w_2 = 0.16$)

Region 3 point estimate ($w_3 = 0.15$)

Posterior mean (point-wise) estimate.

Figure 5: Main results for the "blocks" test function with $N = 2048$ and $\sigma = 2.5$.



Region 4 point estimate ($w_4 = 0.14$) **Region 5 point estimate** ($w_5 = 0.12$)

Region 6 point estimate ($w_6 = 0.09$) **Region 7 point estimate** ($w_7 = 0.08$)

Region 8 point estimate ($w_8 = 0.07$) **Region 9 point estimate** ($w_9 = 0.03$)

Region 10 point estimate ($w_{10} = 0.01$) **Region 11 point estimate** ($w_{11} < 0.01$)

Region 12 point estimate ($w_{12} < 0.01$) **Region 13 point estimate** ($w_{13} < 0.01$)

Figure 6: Regions 4-13 for $N = 2048$ and $\sigma = 2.5$.

7 Further Work

7.1 Splitting Algorithm

The reassignment phase of the variable dimension posterior algorithm that we have used can only reassign elements to existing regions that were created during the first phase. It could be modified to allow for the birth and death of regions.

7.2 Superfluous Parameters

For each region we have used the Kullback-Leibler distance as a loss function to estimate the point estimate. The point estimate is therefore a good representative in terms of predictive performance for the models contained within a region. However, this method of point estimation does not take into account the number of parameters in the model estimated. This problem was not extreme[5] in our examples because we did not estimate

[5] An example where this issue occurs can be seen in Figure 5 for the last segment of region 2. The last change-point looks to be superfluous.

the parameters for each segment. If we were to do so, and used an infinite posterior sample size, then we would find that the point estimate for each region would contain a change-point between every datum. While this does not affect predictive performance, it does affect the human comprehension property that we require. A general, objective means of achieving parsimony in the number of parameters is an area that requires more investigation.

8 Conclusion

We have discussed the problem of producing a special kind of epitome of a posterior distribution with properties that we call Bayesian Posterior Comprehension (BPC). The epitome breaks down a posterior distribution into a small weighted subset of models from the parameter space. Such a set can be used as point estimates, for human comprehension and for fast approximation of posterior expectations. The Minimum Message Length (MML) instantaneous codebook corresponds to an epitome with BPC properties. A general methodology called Message from Monte Carlo, for constructing instantaneous MML codebooks, was extended and demonstrated on several problems with positive results.

References

J. M. Bernardo and A. F. M. Smith. *Bayesian Theory.* Wiley Series in Probability and Mathematical Statistics. John Wiley and Sons, Chichester, 1994.

D. M. Boulton and C. S. Wallace. A program for numerical classification. *The Computer Journal*, 13(1):63–69, 1970.

D. G. T. Denison, B. K. Mallick, and A. F. M. Smith. Automatic Bayesian curve fitting. *Journal of the Royal Statistical Society. Series B (Methodological)*, (60):335–350, 1998.

A. J. Dobson. *An Introduction to Statistical Modelling.* Chapman and Hall, London, 1983.

D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81:425–455, 1994.

D. L. Dowe, R. A. Baxter, J. J. Oliver, and C. S. Wallace. Point estimation using the Kullback-Leibler loss function and MML. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD98)*, volume 1394 of *Lecture Notes in Artificial Intelligence*, pages 87–95. Springer-Verlag, 1998.

G. E. Farr and C. S. Wallace. The complexity of strict minimum message length inference. *The Computer Journal*, 45(3):285–292, 2002.

L. J. Fitzgibbon, D. L. Dowe, and L. Allison. Message from Monte Carlo. Technical Report 107, School of Computer Science and Software Engineering, Monash University, Clayton, Victoria 3800, Australia, 2002a.

L. J. Fitzgibbon, D. L. Dowe, and L. Allison. Univariate polynomial inference by Monte Carlo message length approximation. In C. Sammut and A. Hoffmann, editors, *Proceedings of the Nineteenth International Conference on Machine Learning (ICML-2002)*, pages 147–154, San Francisco, July 2002b. University of New South Wales, Sydney, Australia, Morgan Kaufmann.

W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. *Markov chain Monte Carlo in practice*. Chapman-Hall, London, 1996.

W. R. Gilks, A. Thomas, and D. J. Spiegelhalter. A language and program for complex Bayesian modelling. *Statistician. Special Issue: Conference on Practical Bayesian Statistics*, 43(1):169–177, 1994.

P. J. Green. Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82:711–732, 1995.

J. K. Lindsey. *Models for Repeated Measurements*. Oxford Statistical Science Series 10. Clarendon Press, Oxford, 1994.

D. M. Madigan and A. E. Raftery. Model selection and accounting for model uncertainty in graphical models using Occam's window. *Journal of the American Statistical Association*, 89(428):1535–1546, December 1994.

A. E. Raftery, D. Madigan, and J. A. Hoeting. Bayesian model averaging for linear regression models. *Journal of the American Statistical Association*, 92:179–191, 1997.

C. S. Wallace and D. M. Boulton. An information measure for classification. *Computer Journal*, 11(2):185–194, August 1968.

C. S. Wallace and D. M. Boulton. An invariant Bayes method for point estimation. *Classification Society Bulletin*, 3(3):11–34, 1975.

C. S. Wallace and D. L. Dowe. Minimum message length and Kolmogorov complexity. *Computer Journal*, 42(4):270–283, 1999.

C. S. Wallace and P. R. Freeman. Estimation and inference by compact encoding (with discussion). *Journal of the Royal Statistical Society. Series B (Methodological)*, 49: 240–265, 1987.