Change-Point Estimation Using New Minimum Message Length Approximations

Leigh J. Fitzgibbon, David L. Dowe, and Lloyd Allison

School of Computer Science and Software Engineering
Monash University, Clayton, VIC 3168 Australia
{leighf,dld,lloyd}@bruce.csse.monash.edu.au

Abstract. This paper investigates the coding of change-points in the information-theoretic Minimum Message Length (MML) framework. Change-point coding regions affect model selection and parameter estimation in problems such as time series segmentation and decision trees. The Minimum Message Length (MML) and Minimum Description Length (MDL78) approaches to change-point problems have been shown to perform well by several authors. In this paper we compare some published MML and MDL78 methods and introduce some new MML approximations called 'MMLDc' and 'MMLDF'. These new approximations are empirically compared with Strict MML (SMML), Fairly Strict MML (FSMML), MML68, the Minimum Expected Kullback-Leibler Distance (MEKLD) loss function and MDL78 on a tractable binomial change-point problem.

1 Introduction

Change-points can be found in many machine learning problems. They arise where there is a need to partition data into *contiguous* groups which are to be modelled distinctly. The inference of change-points (the boundaries of the contiguous groups) is important since change-points describe a point of transition between different states of stochastic behaviour of the data. They can be used to explain the generating process and also for prediction of future data.

The Minimum Message Length (MML) principle [1–3] is an invariant Bayesian point estimation technique based on information theory. MML selects regions from the parameter space which contain models that can justify themselves with high posterior probability mass [3, page 276]. Using MML we are able to capture the important information in the posterior. For example, the best explanation of the data might be that "a change-point occurred between times t_1 and t_2 and the point estimate that best summarizes this region is $\hat{t}$". The MML method is especially useful when many change-points are being estimated and on large data-sets - for example, segmentation of a DNA string. DNA strings can be very large, containing millions of characters. It would be impractical to deal with a posterior distribution over such strings using contemporary computational techniques.

Previous work on coding change-point parameters in the MML framework has resulted in analytical approximations which treat the change-point as a continuous parameter [4–7] or avoid stating them altogether [8]. These methods work well in practice. However, change-points are realized as discrete parameters since they partition a data sample, and in this paper we investigate new MML approximations which treat them discretely.

The paper proceeds by describing a binomial change-point problem. We then consider the two computationally infeasible MML criteria: Strict MML (SMML) and Fairly Strict MML (FSMML) in Section 3.1 and 3.2. The algorithms to compute the SMML and FSMML codes have exponential time complexity for the binomial problem, which limits the experiments to small samples, but the results still give insight into the behaviour of the methods. We then describe two new approximations called MMLDc and MMLDF. These are practical methods that are motivated by SMML (in part), FSMML and MML87 [2]. In Section 5 we empirically compare these new approximations with SMML, FSMML and other existing methods.

2 Binomial Problem

A Bernoulli trial is conducted with K independent coin tosses. The results are recorded in a binary string x, where $T = 0$ and $H = 1$. It is suspected that the bias of the coin may have changed at some point in time, ϕ, during the trial. Given the data from the trial, we wish to infer the best explanation: was there a change-point and, if so, where was it? We denote the change-point parameter by ϕ and its parameter-space by $\varPhi$. We often speak in terms of the number of groups of data rather than the number of change-points. And, in our notation, we use G for the number of groups ($G = \{1, 2\}$).

The likelihood for the change-point model is:

$$f(x|G) = \begin{cases} f(x) = f_{null}(x) & G = 1 \\ f(x|\phi) = f_L(x_1^{\phi}) f_R(x_{\phi+1}^{K}) & G = 2 \end{cases} \tag{1}$$

where f_{null} is the model for the $G = 1$, no change-point hypothesis; and f_L and f_R are the models for groups to the left and right of the change-point.

The likelihood function for an ordered Bernoulli trial, which we will be using for f_{null}, f_L and f_R is:

$$f_{bin}(x|p) = p^{\sum x_i}(1-p)^{K-\sum x_i} \quad x_i = 0, 1 \tag{2}$$

To make the SMML and FSMML solutions computationally feasible, the experiments are simplified as follows. For $G = 2$, we use a uniform prior over the change-point location (i.e. $h(\phi) = \frac{1}{K-1}$), and we have a uniform prior for the number of change-points (i.e. h(G=1) = h(G=2) = 0.5). The f_{null}, f_L and f_R likelihood functions that we have chosen to use have fixed biases, and therefore have no free parameters. The biases we use are 0.25, 0.15 and 0.75 for f_{null}, f_L and f_R respectively. We use fixed coins to reduce the estimation problem to the two discrete parameters of interest: G and ϕ. This is necessary to make the construction of the SMML and FSMML (code-books and) estimators feasible. However, even though we are using such a simple model there is still an exponential step (see Section 3.1), so experimenting with large amounts of data is not possible.

3 The Minimum Message Length Principle

In the Minimum Message Length (MML) framework [1–3], inference is framed as a coding process. The aim is to construct a code-book that would (hypothetically) allow for the transmission of the data in a two-part message over a noiseless channel as briefly as possible. From coding theory we know that an event with probability p can be encoded in a message with length $-\log_2(p)$ bits using an ideal Shannon code. Using a Bayesian setting, the sender and receiver agree on a prior distribution $h(\theta)$ and likelihood function $f(x|\theta)$ over the parameter space Θ and data-space X. An estimator is a function from the data-space to the parameter-space, denoted $m : X \to \Theta$. After observing some data x, we can use an estimator to construct a two-part message encoding the estimate $\hat{\theta} = m(x)$ in the first part and then the data using the estimate, $x|m(x)$, in the second.

3.1 Strict Minimum Message Length (SMML)

The probability that $m(.)$ returns an estimate $\hat{\theta}$ is $q(\hat{\theta}) = \sum_{x:m(x)=\hat{\theta}} r(x)$, where $r(x)$ is the marginal probability of the data, x. The length of the first part of the message is therefore $-\log q(m(x))$, and the length of the second part of the message is $-\log f(x|m(x))$. The sender and receiver will use the code-book with estimator, $m(.)$, which minimises the expected message length:

$$I_1 = -\sum_{x \in X} r(x) \left(\log q(m(x)) + \log f(x|m(x))\right) \tag{3}$$

The estimator which minimises I_1 is called the Strict Minimum Message Length (SMML) estimator [2, page 242] [9, 10, 3]. The construction of I_1 is NP-hard for most distributions. The only distributions that it has reportedly been constructed for are the binomial and trinomial (trinomial using a heuristic) [9] and $N(\mu, 1)$ [11, page 22].

The construction of SMML estimators is simplified when there exists a sufficient statistic of lesser dimension than the data-space. Unfortunately, for univariate change-point parameters, the minimal sufficient statistics are of the same dimension as the data. Since we therefore cannot reduce the dimensionality of the data-space, we are left with the SMML code-book construction problem of trying to optimally assign the 2^K elements of the data-space to estimates. For the experiments in this paper we use an EM algorithm which randomly selects an element of the data-space and then finds the optimal code-book assignment $\hat{\theta} = m(x)$ using: $\hat{\theta} = argmax_{\theta \in \Theta} \left[q(\theta) f(x|\theta)\right]$.

This is not guaranteed to minimise I_1 since the algorithm can easily get stuck in local optima. To try and avoid this, we iterate the SMML algorithm a number of times with and without seeding the algorithm with the FSMML partition discussed in the next section. The resulting algorithm still has exponential time complexity. The SMML estimates for up to $K = 15$ can be seen in Figure 1. The bold dots in the diagram illustrate the point estimates that are used in the code-book. We can see that for up to, and including, $K = 7$ the estimator always infers that there was no change-point. As the data-space gets larger, point estimates start appearing to the left of the change-point parameter space. This asymmetry is explained by the choice of biases used.

3.2 Fairly Strict Minimum Message Length (FSMML)

The FSMML [12] estimator is an approximation to SMML based on a partition of the parameter space. The FSMML expected message length is:

$$I_{1a} = -\sum_{\hat{\theta}\in\Theta^*} q(\hat{\theta}) \log q(\hat{\theta}) - \sum_{\hat{\theta}\in\Theta^*} \int_{\theta\in s(\hat{\theta})} h(\theta) \sum_{x\in X} f(x|\theta) \log f(x|\hat{\theta})\, d\theta \qquad (4)$$

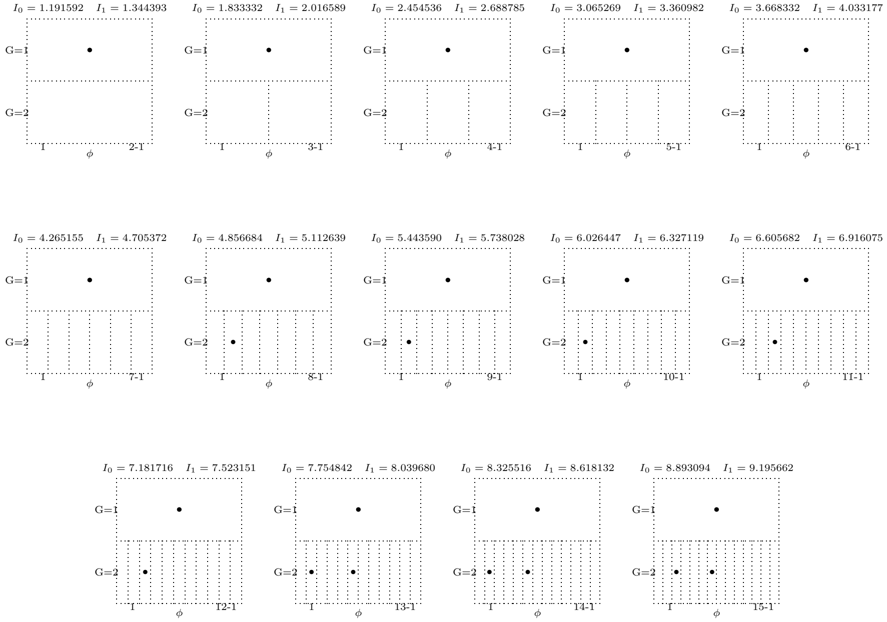
where $q(\hat{\theta})$ is approximated as $q(\hat{\theta}) = \int_{\theta\in s(\hat{\theta})} h(\theta)\, d\theta$, Θ^* is the set of point estimates, and $s(\hat{\theta})$ is the region of the parameter-space which is grouped with point estimate $\hat{\theta}$.

We minimise I_{1a} by searching for the optimal partition of the parameter-space and the $\hat{\theta}$ for each segment of the partition. Since I_{1a} consists of a sum over independent partitions, we can use W. D. Fisher's [13] polynomial time dynamic programming algorithm[1]. We therefore seek the partition of change-points and the estimates which minimise I_{1a}. We allow the partition to contain models from different subspaces since all we are attempting to do is group similar models in such a way that minimises the expected two-part message length.

The algorithm we use is guaranteed to find the optimal solution. It consists of a high-order polynomial step to find the partition (using Fisher's algorithm), and an exponential step to compute a revised version of the message length (I_{1b} in the Figures). The FSMML partitions for up to $K = 15$ can be seen in Figure 2. The partition is represented by the solid shapes, and the bold dots represent the point estimates used in each region. We can see that once K is greater than eight, the partitions consist of models from different subspaces. This allows data generated by change-points on the right to be modelled by the no change-point model. What the FSMML partition is saying is that we cannot reliably distinguish between models in this region, and they are best modelled with the no change-point model. In the figures, the FSMML code-book looks very similar to the SMML code-book. However, as expected, for many values of K, the SMML estimator has a slightly better message length. This is because it is able to individually assign elements of the data-space to estimates.

[1] This is the same algorithm that has been used for partitioning the data-space in the SMML binomial case [9].

Fig. 1. SMML Estimates K = 2..15



$I_0 = 1.191592$ $I_1 = 1.344393$

$I_0 = 1.833332$ $I_1 = 2.016589$

$I_0 = 2.454536$ $I_1 = 2.688785$

$I_0 = 3.065269$ $I_1 = 3.360982$

$I_0 = 3.668332$ $I_1 = 4.033177$

$I_0 = 4.265155$ $I_1 = 4.705372$

$I_0 = 4.856684$ $I_1 = 5.112639$

$I_0 = 5.443590$ $I_1 = 5.738028$

$I_0 = 6.026447$ $I_1 = 6.327119$

$I_0 = 6.605682$ $I_1 = 6.916075$

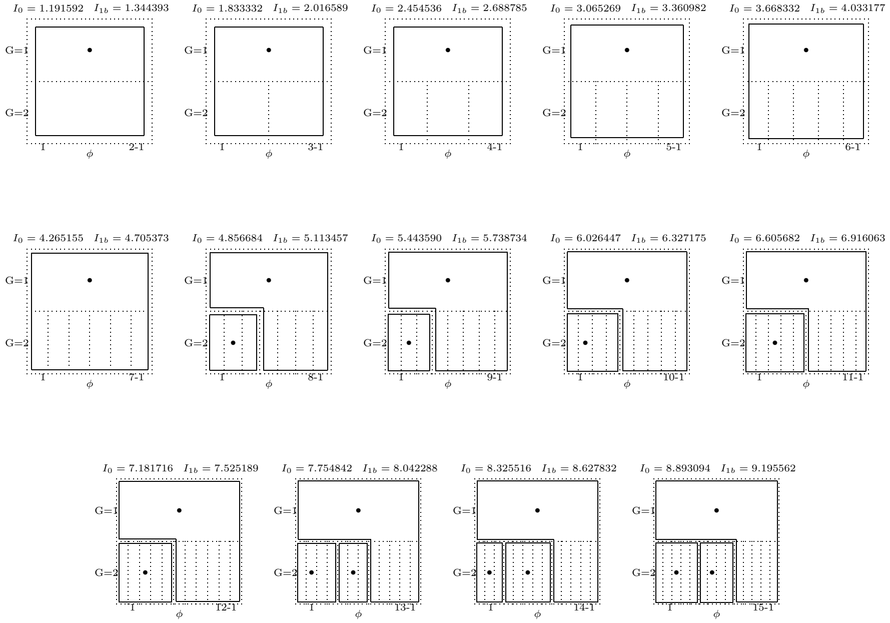
$I_0 = 7.181716$ $I_1 = 7.523151$

$I_0 = 7.754842$ $I_1 = 8.039680$

$I_0 = 8.325516$ $I_1 = 8.618132$

$I_0 = 8.893094$ $I_1 = 9.195662$

Fig. 2. FSMML Partitions K = 2..15



$I_0 = 1.191592$ $I_{1b} = 1.344393$

$I_0 = 1.833332$ $I_{1b} = 2.016589$

$I_0 = 2.454536$ $I_{1b} = 2.688785$

$I_0 = 3.065269$ $I_{1b} = 3.360982$

$I_0 = 3.668332$ $I_{1b} = 4.033177$

$I_0 = 4.265155$ $I_{1b} = 4.705373$

$I_0 = 4.856684$ $I_{1b} = 5.113457$

$I_0 = 5.443590$ $I_{1b} = 5.738734$

$I_0 = 6.026447$ $I_{1b} = 6.327175$

$I_0 = 6.605682$ $I_{1b} = 6.916063$

$I_0 = 7.181716$ $I_{1b} = 7.525189$

$I_0 = 7.754842$ $I_{1b} = 8.042288$

$I_0 = 8.325516$ $I_{1b} = 8.627832$

$I_0 = 8.893094$ $I_{1b} = 9.195562$

3.3 MML68 Change-Point Approximation

Oliver, Baxter and co-workers have applied the MML68 [1] estimator methodology to the segmentation problem with Gaussian segments [4] [11, chapter 9] [5, 6]. They have derived MML formulas for stating the change-point locations to an optimal precision independently of the segment parameters. The same method has been used [7] for the problem of finding change-points in noisy binary sequences [14] - where it compared favourably with Akaike's Information Criterion (AIC), Schwarz's Bayesian Information Criterion (BIC), an MDL-motivated metric of Kearns et al. [14] and a more correct version of Minimum Description Length[7].

We apply the MML68 approximation to the binomial problem in this paper. Assuming that the true change-point is uniformly distributed in some range of width, R, we encode the data using the point estimate $\hat{\phi}$ at the centre of this region. The true change-point is equally likely to be to the right or to the left of the point estimate. If it is located to the right then its expected value is $\hat{\phi} + \frac{R}{4}$, and if it is located to the left its expected value is $\hat{\phi} - \frac{R}{4}$. The expected message length is computed by averaging the expected coding inefficiency of these two scenarios which for our Bernoulli problem simplifies to an expression involving the Kullback-Leibler distance, $KL(.||.)$:

$$MessLen \approx -\log(\frac{R}{K-1}) + \frac{R}{8}\left(KL(p_R||p_L) + KL(p_L||p_R)\right) - \log f(x|\hat{\phi}) \quad (5)$$

where p_L and p_R correspond to the distributions of the coins to the left and right of the change-point respectively. Using Equation 5, the size of the region which minimises the message length is easily derived.

4 A New Approximation to FSMML: MMLD

Minimum Message Length approximation D (MMLD) can be thought of as a numerical approximation to FSMML. It was proposed by D. L. Dowe and has been investigated by his student [15]. MMLD is based on choosing a region R of the parameter space after observing some data. It was partly motivated by improving the Taylor expansion approximation of MML87 [2] while retaining invariance and, like MML87, avoids the problem of creating the whole code-book, which would typically require enumeration of the data and parameter spaces in SMML and FSMML. Given an uncertainty region, R, MMLD approximates the length of the first part of the message as the negative log integral of the prior over R (like FSMML). The length of the second part is approximated by the expected value (with respect to the prior), over R, of the negative log-likelihood. This gives rise to an MMLD message length of

$$MessLen \approx -\log\left(\int_R h(\theta)\, d\theta\right) - \frac{\int_R h(\theta) \log f(x|\theta)\, d\theta}{\int_R h(\theta)\, d\theta} \quad (6)$$

Equation 6 makes no explicit claim about which point estimate should be used to encode data for the region, R. Once the region has been found which minimises it, we need to find a point estimate which summarizes the models in the region. Since the estimates produced by the FSMML estimator are equivalent to the minimum expected Kullback-Leibler distance estimator (the expectation being taken with respect to the prior over the region, rather than the posterior) [12, 10], we have used this for the experiments involving MMLD:

$$\hat{\theta} = argmin_{\hat{\theta} \in R} \int_{\theta \in R} h(\theta) KL(\theta, \hat{\theta}) \, d\theta \tag{7}$$

Whereas FSMML can build code-books consisting of non-contiguous regions (i.e., combine modes or models from different subspaces) with minimum expected message length, MMLD cannot in general. This is because MMLD does not take into account the similarity of the models it combines in R - it only cares about their prior probability and likelihood. If we attempt to build non-contiguous regions, then in variable dimension problems or where the likelihood is multi-modal, MMLD will possibly combine modes. The models contained within these modes may be quite different (i.e., have large Kullback-Leibler distances), or they may be similar (i.e., have small Kullback-Leibler distances). For the latter case, combining modes is a valid thing to do. However, in general, we would expect the models contained in two distinct modes to be quite different and, for *inference*, we risk underestimating the message length if they are grouped into the same region.

So, rather than simply choose the region R to optimise Dowe's MMLD message length expression in Equation 6, Fitzgibbon has suggested that we invoke the FSMML 'Boundary Rule' [12] to determine whether a model should be considered for membership of R. The Boundary Rule is a heuristic used to choose the optimal partition for the FSMML expected message length equation (Equation 4), where a candidate model θ is considered to be a member of the region (with point estimate $\hat{\theta}$ - the minimum prior-weighted expected Kullback-Leibler distance estimate for the region) if the following constraint is satisfied:

$$\theta \in R \text{ iff } KL(\theta, \hat{\theta}) \leq \frac{\int_{\theta \in R} h(\theta) KL(\theta, \hat{\theta}) \, d\theta}{\int_{\theta \in R} h(\theta)} + 1$$

We denote the MMLD approximation augmented by the FSMML Boundary Rule as MMLDF. While other (non-contiguous) versions of MMLD exist, throughout the remainder of this paper, MMLDc will refer to using a contiguous region (i.e. R contains only models of the same dimension and from a single mode). We include both MMLDc and MMLDF in the experiments to compare the advantage of allowing the region to consist of models from different subspaces. For the binomial problem with known biases, the parameter space is discrete - so an exhaustive search for the optimal region was performed.

5 Empirical Comparison and Discussion

Compact coding methods attempt to minimise the expected length of a two-part message. However, they cannot be judged on this criterion since - other than SMML and FSMML - the methods only approximate the message length. Furthermore, we are not really interested in creating short messages per se but rather in how good the inferred statistical model is. The definition of a good model will depend on what use the model will be put to. We therefore use the following general criteria: the Kullback-Leibler (KL) distance between the true and inferred models; and the mean squared error in estimation of the change-point location (if it exists). We have compared the MMLD approximation both with and without the FSMML Boundary Rule (MMLDF and MMLDc respectively) with SMML, FSMML, MML68 (as described in Section 3.3), Minimum Description Length (MDL78) [16] and the Minimum Expected KL Distance loss function (MEKLD) [10]. We ran 10^4 trials for each $K = 2..15$ where we sampled from the prior and then generated data. Each method was given the data and the biases of the coins used to generate the data and then asked to infer whether or not a change-point occurred and, if so, where it was located.

We have plotted the average KL distance for each method in Figure 3. The SMML and FSMML estimators had significantly higher KL distances than the other methods for $K > 4$. MEKLD had the lowest on average, as expected. MML68 performed well and was not far behind MEKLD. Our MMLDF estimator was close behind MML68 and slightly better than our MMLDc estimator.

Figure 4 shows the average squared error in estimating the change-point location for each method. The average is taken over the instances where the method correctly inferred that there was a change-point. The SMML and FSMML estimators performed exceptionally well. Their good performance here and poor KL distance performance indicates that they prefer not to infer a change-point unless they are reasonably certain of its location. The MMLDF estimator comes second to SMML and FSMML for $K < 13$. The MML68 method, which had very good KL distance performance, performed poorly for this criterion.

We note that MMLDF outperforms MMLDc for both criteria, therefore providing evidence that building non-contiguous coding regions - which SMML theory and FSMML theory both advocate - is advantageous. The MMLDF estimator appears to be robust and has good explanatory (i.e., has small squared error in change-point location when correctly inferring change-points) and predictive powers (i.e., has small KL distances).

6 Conclusion

We have empirically compared a number of information-theoretic methods for estimating change-points including two new Minimum Message Length approximations. The comparison was based on a binomial problem using small sample sizes which allowed us to include the computationally impractical Strict MML (SMML) and Fairly SMML (FSMML) estimators. In the comparison we found that the performance of the MMLDc approximation was improved by incorporating the Kullback-Leibler Boundary Rule, therefore allowing coding regions to contain models from different subspaces (MMLDF) whilst still approximating an efficient FSMML code-book. MMLDF was robust and performed well in terms

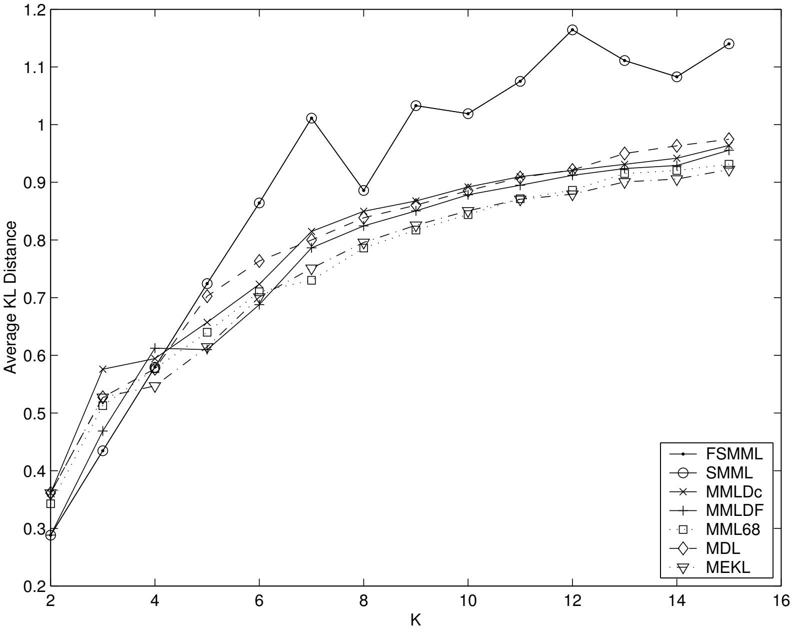


Fig. 3. Average Kullback-Leibler Distance (10^4 trials)

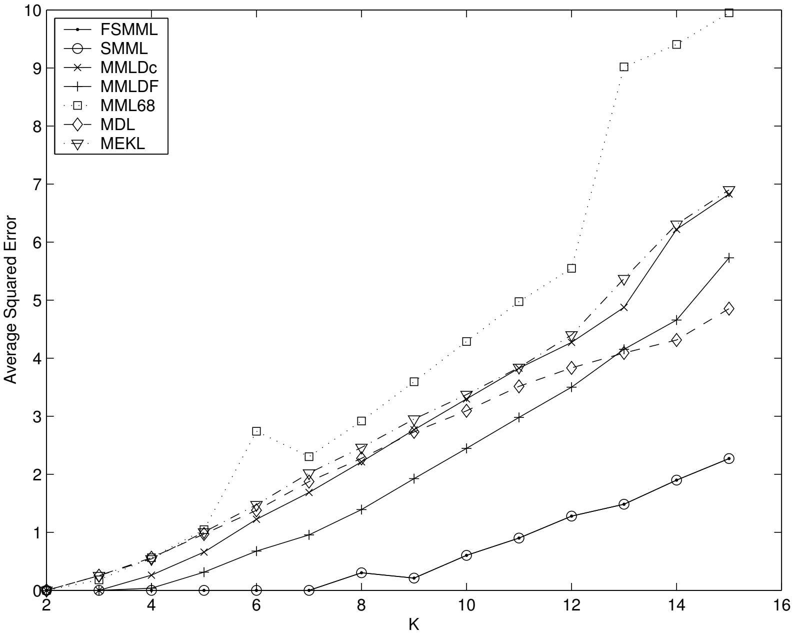


Fig. 4. Average Squared Error of Change-Point Location (10^4 trials)

of Kullback-Leibler distance and (squared) error in estimation of the change-point location (where inferred). Use of MMLD and variations for more difficult problems will be investigated in forthcoming work.

References

1. Wallace, C.S., Boulton, D.M.: An information measure for classification. Computer Journal **11** (1968) 185–194
2. Wallace, C.S., Freeman, P.R.: Estimation and inference by compact encoding (with discussion). Journal of the Royal Statistical Society. Series B (Methodological) **49** (1987) 240–265
3. Wallace, C.S., Dowe, D.L.: Minimum message length and Kolmogorov complexity. Computer Journal **42** (1999) 270–283
4. Baxter, R.A., Oliver, J.J.: The kindest cut: minimum message length segmentation. In Arikawa, S., Sharma, A.K., eds.: Proceedings of the Seventh International Workshop on Algorithmic Learning Theory. Volume 1160 of LNCS., Springer-Verlag Berlin (1996) 83–90
5. Oliver, J.J., Forbes, C.S.: Bayesian approaches to segmenting a simple time series. Technical Report 97/336, Department Computer Science, Monash University, Australia 3168 (1997)
6. Oliver, J.J., Baxter, R.A., Wallace, C.S.: Minimum message length segmentation. In Wu, X., Kotagiri, R., Korb, K., eds.: Research and Development in Knowledge Discovery and Data Mining (PAKDD-98), Springer (1998) 83–90
7. Viswanathan, M., Wallace, C.S., Dowe, D.L., Korb, K.: Finding cutpoints in noisy binary sequences - a revised empirical evaluation. In: Australian Joint Conference on Artificial Intelligence. (1999)
8. Fitzgibbon, L.J., Allison, L., Dowe, D.L.: Minimum message length grouping of ordered data. In Arimura, H., Jain, S., eds.: Proceedings of the Eleventh International Conference on Algorithmic Learning Theory (ALT2000). LNAI, Springer-Verlag Berlin (2000) 56–70
9. Farr, G.E., Wallace, C.S.: Algorithmic and combinatorial problems in strict minimum message length inference. In: Research on Combinatorial Algorithms. (1997) 50–58
10. Dowe, D.L., Baxter, R.A., Oliver, J.J., Wallace, C.S.: Point estimation using the Kullback-Leibler loss function and MML. In: Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD98). Volume LNAI of 1394., Springer-Verlag (1998) 87–95
11. Baxter, R.A.: Minimum Message Length Inductive Inference: Theory and Applications. PhD thesis, Department of Computer Science, Monash University (1996)
12. Wallace, C.S.: PAKDD-98 Tutorial: Data Mining. Monash University, Australia (Book in preparation) (1998)
13. Fisher, W.D.: On grouping for maximum homogeneity. Journal of the American Statistical Society **53** (1958) 789–798
14. Kearns, M., Mansour, Y., Ng, A.Y., Ron, D.: An experimental and theoretical comparison of model selection methods. Machine Learning **27** (1997) 7–50
15. Lam, E.: Improved approximations in MML. Honours thesis, Monash University, School of Computer Science and Software Engineering, Monash University, Clayton, Australia (2000)
16. Rissanen, J.J.: Modeling by shortest data description. Automatica **14** (1978) 465–471