Advance Access publication on January 27, 2008
doi:10.1093/comjnl/bxm121

Wallace's Approach to Unsupervised Learning: The Snob Program

Murray A. Jorgensen[1,*] and Geoffrey J. McLachlan[2]

[1]*Department of Statistics, University of Waikato, Hamilton, New Zealand*
[2]*Department of Mathematics, University of Queensland, St. Lucia, Brisbane, Qld 4072, Australia*
**Corresponding author: maj@stats.waikato.ac.nz*

We describe the Snob program for unsupervised learning as it has evolved from its beginning in the 1960s until its present form. Snob uses the minimum message length principle expounded in Wallace and Freeman (Wallace, C.S. and Freeman, P.R. (1987) Estimation and inference by Compact coding. J. Roy. Statist. Soc. Ser. B, 49, 240–252.) and we indicate how Snob estimates class parameters using the approach of that paper. We will survey the evolution of Snob from these beginnings to the state that it has reached as described by Wallace and Dowe (Wallace, C.S. and Dowe, D.L. (2000) MMM mixture modelling of multi-state, Poisson, Von Mises Circular and Gaussian distributions. *Stat. Comput.*, 10, 73–83.) We pay particular attention to the revision of Snob in the 1980s where definite assignment of things to classes was abandoned.

Keywords: *computer program, EM algorithm, minimum message length, mixture model, cluster analysis*

1. INTRODUCTION

In this article, we consider the work of Chris Wallace, his students and collaborators, in unsupervised learning. This area is also known as clustering, cluster analysis or numerical taxonomy. We focus our attention on the pioneering Snob program, wryly so-called because it places individuals in classes (C.S. Wallace, personal communication).

The Snob program [1, 2] represents a pioneer contribution to a model-based approach to unsupervised learning. At the same time, as Snob made an early contribution to model-based clustering (as unsupervised learning based on probability distributions for the clusters), it also provided early evidence that a form of inference based on coding theory could tackle non-trivial applications involving substantial amounts of data.

Other approaches to clustering that appeared at approximately the same time in the statistical literature [3–9] also considered a mixture model-based approach, primarily focussing on the assumption of normality for the component distributions. In a related approach, Hartley and Rao [10] and Scott and Symons [11] considered the so-called classification—likelihood method of clustering.

As to be discussed later in more detail, the distinction between the mixture and classification approaches to clustering is on how they are formulated and subsequently implemented. Both work with the joint likelihood formed on the basis of the observed data and the unobservable class-indicator variables. However, with the classification approach, these indicator variables are treated as unknown parameters to be estimated along with the unknown parameters in the assumed distributional forms for the class densities corresponding to the clusters to be imposed on the data. In contrast, with the mixture approach via the EM algorithm, these class indicator variables are treated as 'missing data' and at each iteration of that algorithm are replaced by the current values of their conditional expectations, given the observed data. The mixture approach thus circumvents the biases in the parameter estimates produced by the classification approach due to outright (hard) assignments of the observations during the iterative process. The Snob program initially used hard assignments, but later switched to soft (partial) assignments in its implementation of the minimum message length (MML) approach.

2. AN INFORMATION MEASURE

Wallace and Boulton [1] do not present their work as a new method for grouping cases into classes but rather as a criterion for evaluating the results of such clusterings, as they state in their introduction:

> The aim in this paper is to propose a measure of the goodness of a classification, based on information theory, which is completely independent of the process used to generate the classification.

Of course, once such a criterion is proposed, it is a small step to seek a clustering that optimizes it. That there was and is a need for such a criterion cannot be denied. Consulting works such as [12] or [13] reveals an embarrassment of clustering methods and the alternatives have only increased with the passage of time. Consider, for example, traditional hierarchical clustering methods based on a distance or similarity matrix. The method of calculating the distance or similarity measure must be specified, as must the criterion used for combining or dividing classes, and the level of similarity at which the resulting dendrogram is cut. Each of these decision points offer a profusion of choices, which multiply up to give the final number of available methods.

The measure that Wallace and Boulton [1] proposed will later be developed into a more general context as the principle of MML. They considered a digital encoding of a message the purpose of which is to describe the attribute values of each observation. A useful classification will divide all the observations into a finite number of concentrated classes. This usefulness is reflected in the coding scheme by allowing a short encoding of observations within a class. In their words

> If the expected density of points in the measurement space is everywhere uniform, the positions of the points cannot be encoded more briefly than by a simple list of the measured values. However, if the expected density is markedly non-uniform, application of Shannon's theorem will allow a reduction in the total message length by using a brief encoding for positions in those regions of high expected density.

The criterion for evaluating a classification of points into classes will be the length of a message that describes all the attribute values of all the data points that are constructed with the assistance of the classification. The message is divided into five parts communicating (i) the number of classes; (ii) a dictionary of class names; (iii) a description of the distribution function for each class; (iv) for each point, the name of the class to which it belongs and (v) for each point, its attribute values in the code set up for its class.

We remark parenthetically that in all his writings on classification, Wallace has eschewed terms like 'observation', 'case' or 'operational taxonomic unit' in favour of the pithy 'thing'. It is regrettable that this sensible lead was not followed by most of the literature.

We refer the reader to the original paper for the details of the message construction, but we will make brief comments on the different components of the message.

2.1. Number of classes

Wallace and Boulton [1] effectively assume an equal probability for any number of classes up to some arbitrary cut-off. This means a constant length for this part of the message, which is therefore disregarded.

2.2. Class name dictionary

The receiver of the message is assumed to be in possession of a 'code book' containing a number of possible sets of class names. This part of the message will tell the receiver which set of names will be used in the message. If the code book contains a large number of sets of names, then it will be possible for the sender to select one such set in which the large classes receive short names, which will be useful when sending the class name for every observation. However, too large a code book means that the part of the message that specifies which set to use must itself be very large. Balancing these two requirements leads to a fairly long technical discussion in the appendix to [1].

2.3. Description of the class distribution function

It is assumed that within each class, the attributes are independently distributed. This permits the multivariate distribution to be described by simply concatenating the descriptions of the distributions of each attribute. (Recent versions of Snob relax this requirement.) The considerations for encoding the distribution of a categorical attribute are similar to those discussed for the class name dictionary. The encoding of a continuous variable within a class is based on the assumption of a normal distribution for the variable. Decisions must be made on the values of mean and variance to state, and the precision with which to state these. This leads to another technical section of the paper to determine approximately optimal values.

2.4. Class and attribute values for each observation

These merely need to be stated in the coding schemes described earlier in the message.

2.5. The form of the message

Putting all the components of the message together leads to a large and not particularly elegant expression for message length. While a drawback for analytical work, this is not a particular problem for comparing the message lengths corresponding to various proposed clusterings of a data set, as the lengths in any case would in practice be computed by a computer program.

3. THE SNOB PROGRAM

The Snob program itself gets only a brief mention by Wallace and Boulton [1] as a program that attempts to minimize the measure defined in the paper. We need to refer to [2] for a description of the structure of Snob itself.

Beginning from a message length defined by an initial classification, Snob employs a number of 'tactics' by which the classification is improved, so that it has a reduced message length.

3.1. Distribution adjustment

The number of classes and the assignment of observations to classes are left unchanged, and it is sought to optimize the parameters of the class distributions and the proportion parameters for the classes.

3.2. Reclassifying

The number of classes, the proportion of observations assumed to be in each class and the class distribution parameters are held constant, and the observations are reassigned to their most probable class.

3.3. Splitting

A single class is split into two, and the optimal proportion and distribution parameters are determined for the new classes.

3.4. Merging

Two classes are combined and the optimal proportion and distribution parameters are determined for the new class.

3.5. Swapping

A class is split into two, and one of its parts is added to another class and the optimal proportion and distribution parameters are determined for the affected classes.

The distribution adjustment process is an MML estimation applied separately to each class, as it is currently constituted.

The reclassifying process proceeds observation-by-observation. For each observation, it works out the message length for describing the attribute values of the observation according to the encoding for each class. It assigns the observation to the class for which this length is smallest.

Snob considers each class to be divided into two subclasses. If the program were hypothetically stopped at a T-class solution, there would also be information about a $2T$-class solution. In the splitting procedure, all possible $T+1$-class solutions generated by splitting one class are evaluated in terms of the consequent reduction in message length. The best choice is made and the two subclasses of the chosen class are promoted to full classes and endowed with randomly chosen subclasses (at the following iteration).

Merging brings two classes together, and the distribution adjustment procedure is carried out for the new class. The old classes become the two subclasses of the new class.

In the swapping process one of the four subclasses of the two classes is made into a full class and gets two random subclasses. The other three subclasses form a new class with subclasses given by the transferred subclass and the old class.

Snob is initialized either by starting from an initial classification, which will then be improved in terms of Snob's message length criterion, or by a random start with a given number of classes.

A feature of Snob added at the revision described in [14] is similar to attribute selection, but more flexible. A facility is provided whereby an attribute may be declared 'significant for a class' and distributional parameters estimated specifically for that class, but a common form of the attribute's distribution is assumed for those classes in which it is not declared significant. A class by default, must have existed for five iterations before attributes are tested to see whether being made insignificant for that class results in a shortened message length.

4. PARTIAL OR FULL ASSIGNMENT OF OBSERVATIONS TO CLASSES?

The approach of Snob to unsupervised learning, as noted above, involves the fitting of finite mixtures of probability distributions to data, more briefly: mixture modelling.

There are two ways in which this kind of model may be formalized. Let y_i, $i = 1, \ldots, S$, be the collection of *things* or observations. Each y_i is a vector of D attributes $y_{i1}, y_{i2}, \ldots, y_{iD}$ and may belong to one of T classes.

In the first formalism we write the probability density of y_i as

$$f(y_i) = \sum_{j=1}^{T} \pi_j \, f(y_i; \phi_j),$$

where the proportion parameters π_j sum to 1.

In the second formalism, we introduce ST additional binary parameters $z_{ij} \in \{0,1\}$ with $\sum_j z_{ij} = 1$ and write the probability density of y_i as

$$f_C(y_i) = \prod_{j=1}^{T} \pi_j^{z_{ij}} f(y_i; \phi_j)^{z_{ij}},$$

where the proportion parameters π_j sum to 1 as before. In this second formalism we interpret z_{ij} by

$$z_{ij} = \begin{cases} 1 & \text{if } y_i \in \text{Class}_j, \\ 0 & \text{otherwise.} \end{cases}$$

In the first formalism, when the model is fitted to data by MML, maximum likelihood or some other optimization criterion, we will speak of *partial assignment*; in corresponding situations for the second formalism, we speak of *full assignment*. This is because fitting under the second formalism involves making a specific choice of class for each observation. Under partial assignment no class is definitely specified for an observation, although its membership probabilities in the S classes may be worked out by a Bayes rule calculation and may result in one class being strongly favoured.

When the EM algorithm [15] is adopted to fit mixture models with full assignment by maximum (possibly penalized) likelihood, real-valued quantities similar to z_{ij} are introduced and estimated, but these are not actual model parameters.

Scott and Symons [11] note that many classical cluster analysis methods for observations with continuous attributes can be seen as mixture modelling with full assignment. Banfield and Raftery [16] describe a program for clustering with full assignment that builds on the work of Scott and Symons [11]. McLachlan and Basford [17, p. 31–35] discuss maximum likelihood estimation under the full assignment mechanism under the name *classification likelihood*, citing earlier literature and drawing attention to the presence of bias in the distributional parameters ϕ_j when estimated this way.

It is not difficult to see why full assignment leads to bias in the distributional parameters. Consider a mixture of two univariate normal distributions in similar proportions where both distributions have equal scale parameters. Suppose that the two distributions substantially overlap. In this situation under full assignment, there will be a critical value k such that all observations greater than k are assigned to one distribution, and all observations less than k are assigned to the other. Thus the lower distribution loses its upper tail and the upper distribution loses its lower tail. The separation of the means is exaggerated and the variance of each distribution is underestimated.

As the early version of Snob fully assigned observations to classes, it is subject to this sort of bias. For this reason, Snob was revised to work under partial assignment. In section 6.8.2 of [18] Wallace also considers a mixture of two univariate normal distributions to show that full assignment leads to inconsistent parameter estimates.

5. PARTIAL ASSIGNMENT FOR SNOB

Wallace [14] describes some revisions to Snob which changed Snob from a full assignment to a partial assignment clustering program. This was done in order to avoid the bias problems associated with full assignment. Wallace [14] shows that with partial assignment a shorter message length can be obtained than for full assignment, giving partial assignment an MML justification. The technique is known in the MML community as the *coding trick*.

The ingenious construction that is carried out reorganizes only parts 5 and 6 of the message as described in Section 2. Initially we reorder these so that the encoded class and attribute values for each observation are together. Wallace [14] describes how to proceed next, and we repeat this now with only light editing.

Consider the message segment encoding the class and attribute values of a particular observation which is not the last such segment to be encoded in the message. According to the choice of class for each observation there are T ways of encoding the class and attribute values. Let the lengths of the several possible code segments be $l_1, \ldots, l_T$.

Define

$$p_j = 2^{-l_j}, \quad j = 1, \ldots, T.$$

These p_j values may be identified with the probabilities of getting the data by each of the several mutually-exclusive routes, all consistent with the mixture model.

Define

$$P = \sum_{j=1}^{T} p_j \qquad \text{and} \qquad q_j = \frac{p_j}{P}, \quad j = 1, \ldots, T.$$

To choose the encoding for the data segment, first construct according to some standard algorithm a Huffman code optimized for the discrete probability distribution $\{q_j\colon j = 1, \ldots, T\}$. Note that this distribution is the Bayes posterior distribution over the mutually exclusive routes, given the model and the data segments. From the standard theory of optimal codes, the length m_j of the code word in this Huffman code for route j will be $m_j = -\log q_j$, the code will have the prefix property, and every sufficiently long binary string will have some unique word of the code as its prefix. Now examine the binary string encoding the remainder of the data, that is, the data following the segment being considered. This string must begin with some word of the Huffman code, say the word for route k. Then encode the data segment using route k, hence using a code segment of length l_k. Then the first m_k bits of the binary string for the remainder of the data need not be included in the explanation, as they may be recovered by a receiver after decoding the present data segment.

Consider the net length of the string used to encode the data segment, that is, the length the string used minus the length which need not be included for the remaining data. The net length is

$$\begin{aligned} l_k - m_k &= -\log p_k + \log q_k \\ &= -\log(p_k/q_k) \\ &= -\log P \\ &= -\log\left(\sum_{j=1}^{T} p_j\right) \end{aligned}$$

Merely choosing the shortest of the possible, encodings for the data segment would give a length of

$$-\log\left(\operatorname*{Max}_{j=1}^{T} p_j\right).$$

The coding device, therefore, has little effect when one possible coding is much shorter (more probable a posteriori) than the rest, but can shorten the explanation by as much as log T if they are all equally long.

Still following [14] but less closely, we note that the net length of the message describing the data is the same as would be obtained by assigning no observations to classes and using the mixture density

$$f(y_i) = \sum_{j=1}^{T} \pi_j f(y_i;\, \phi_j)$$

directly to encode the attribute values. However, direct optimization of the mixture density is difficult.

The coding trick uses the following part of the message in order to select which of the T classes is used to do the encoding of the attribute values for the observation being considered. As that code segment has nothing to do with the current observation, it is like making a random choice of the class used with probability given by the posterior class probability for that observation (as that is how the classes are encoded).

If the above procedure were used directly to assign observations to clusters, there would be some similarity between Snob and the stochastic EM method [19, 20]. However instead the $\{q_j\}$ for the ith thing are used to define weights w_{ij} and the 'distribution adjustment' for the jth class is carried out with all data but with weights w_{ij}. (This is not entirely obvious from [14] but is discussed in [21].) The 'coding trick' for MML estimation of mixture models and the reason why it leads to a form of the EM algorithm are again expounded by Wallace in Section 6.8.3 of [18].

6. THE EM ALGORITHM FOR MIXTURE MODELS

Outside the MML community mixture models such as

$$f(y_i) = \sum_{j=1}^{T} \pi_j f(y_i;\, \phi_j)$$

are commonly fitted by maximum likelihood using the EM algorithm (see, e.g. [22]). Here we seek to maximize the likelihood

$$L(\theta) = \prod_{i=1}^{S}\left[\sum_{j=1}^{T} \pi_j f(y_i;\, \phi_j)\right],$$

where θ is the vector of unknown parameters, containing the mixing proportions π_j and the component parameters ϕ_j for $j = 1, \ldots, T$. With respect to this likelihood, we may define the observed information matrix $I(\theta;\, y)$ to be the is the negative Hessian of the log-likelihood for θ evaluated at the data vector y and the parameter vector θ. The expected or Fisher information matrix $F(\theta)$ is then defined by

$$F(\theta) = E_\theta[I(\theta;\, y)].$$

In the EM approach, it is also common to introduce the class assignment indicator variables z_{ij}, $i = 1, \ldots, S$, $j = 1, \ldots, T$ considered above. These are not observed but a function

$$L_C(\theta) = \prod_{i=1}^{S} f_C(y_i) = \prod_{i=1}^{S}\prod_{j=1}^{T} \pi_j^{z_{ij}} f(y_i;\, \phi_j)^{z_{ij}}$$

is introduced that would be a likelihood function if z_{ij} had been observed. Note that $\ell_C(\theta) = \log L_C(\theta)$ splits into a part involving the T proportion parameters π and, for each class j, a part involving the parameters ϕ_j. Bayesian maximum posterior estimation with prior $h(\theta)$ may be accomplished just as easily if log $h(\theta)$ also decomposes in a corresponding way.

The EM algorithm proceeds iteratively from initial estimates for the parameters θ. Each iteration comprises two steps: a step involving an expectation (the E-step) and a step involving a maximization (the M-step).

In the E-step we take the conditional expectation of the z_{ij} given the data and the current parameters obtaining

$$q_{ij} = \frac{\pi_j f(y_i;\, \phi_j)}{\sum_{j=1}^{T} \pi_j f(y_i;\, \phi_j)}.$$

In the M-step $\ell_C(\theta)$, with the z_{ij} replaced by the q_{ij}, is maximized with respect to the parameters, and the maximizing values of π and the ϕ_j become the updated parameter estimates.

The new π_j are thus given by

$$\pi_j = \frac{\sum_{i=1}^{n} q_{ij}}{n}$$

and for $j = 1, \ldots, T$ the new ϕ_j are obtained by T separate weighted maximum likelihood estimations in which the data y_i with weight q_{ij} come from the distribution $f(y_i\,;\phi_j)$. Maximum posterior estimation may be obtained similarly if the log prior splits in the way mentioned above. In this case, we would estimate the $T+1$ parameter set by $T+1$ separate weighted maximum posterior estimations.

With respect to L_C, referred to as the *complete-data likelihood*, we may define the complete-data observed information matrix $I_C(\theta; y, z)$ and the complete-data expected information matrix $F_C(\theta)$, in the usual way. We can also define the complete-data conditional expected information matrix

$$\mathcal{I}_C(\theta; y) = E_\theta[I_C(\theta; y, z)|y].$$

7. THE WALLACE–FREEMAN APPROACH TO INFERENCE

Wallace and Freeman [23] present what seems to be the most comprehensive approach to MML inference published prior to [18]. They motivate and present the following estimate of message length:

$$-\log h(\theta) + \frac{1}{2}\log|F(\theta)| - \log f(y;\,\theta) + \frac{1}{2}n_p \log \kappa_{n_p} + \frac{1}{2}n_p, \qquad (1)$$

where $h(\theta)$ is a prior distribution for the parameter values, $F(\theta)$ the expected (Fisher) information matrix, $f(y;\theta)$ the likelihood function, n_p the number of parameters being estimated and κ_n the n-dimensional optimal quantizing lattice constant ([24], table 2.3]).

The approximated expected message length (1) is very similar to the negative of the following approximation to the logarithm of the integrated likelihood, $\log g(y) = \log \int h(\theta) f(y;\,\theta)\,\mathrm{d}\theta$, obtained by Laplace's method [22, section 6.9.2].

$$\log g(y) \approx$$

$$\log f(y;\,\tilde{\theta}) + \mathit{log}\, h(\tilde{\theta}) - \frac{1}{2}\log|H(\tilde{\theta})| + \frac{1}{2}n_p \log(2\pi). \qquad (2)$$

Here $\tilde{\theta}$ is the posterior mode, and $H(\tilde{\theta})$ is the negative Hessian of the log-posterior for θ evaluated at $\theta{=}\tilde{\theta}$.

An important variant on (2) is

$$\log g(y) \approx$$

$$\log f(y;\,\hat{\theta}) + \mathit{log}\, h(\hat{\theta}) - \frac{1}{2}\log|I(\hat{\theta};\,y)| + \frac{1}{2}n_p \log(2\pi). \quad (3)$$

where the posterior mode is replaced by the MLE $\hat{\theta}$ and $H(\hat{\theta})$ is replaced $I(\hat{\theta}; y)$, the observed information matrix evaluated at $\hat{\theta}$. The right-hand side of Equation (3) is termed the *Laplace empirical criterion* (LEC) by McLachlan and Peel [22]. The apparent similarity of the LEC and MML criteria is confirmed by a recent study [25], which found for a simulation study involving generalized Dirichlet mixtures that MML and LEC performed similarly in determining the number of clusters and better than the other alternatives considered.

Expression (1) is unable to be used directly in Snob because the expected information matrix $F(\theta)$ is very difficult to calculate for mixture models. Even its commonly used approximation, the observed information matrix, $I(\hat{\theta}; y)$, is difficult to obtain for mixture models. The EM algorithm may not be adapted in a way similar to the way it can be adapted for maximum posterior estimation because $\log|F(\theta)|$ cannot be written as the sum of $T+1$ parts each involving only one of the parameter subsets π and the ϕ_j for $j = 1, \ldots, T$.

In Snob (1) is not applied directly to the whole mixture model, but, in a weighted form, to the individual models for each component. This is similar to approximating the Fisher information matrix $F(\theta)$ by the complete-data information matrix $F_C(\theta)$ [26].

We note that the determinants of the complete-data expected information matrix $F_C(\theta)$ and the (incomplete data) information matrix $F(\theta)$ can be quite different. To see this note the rate of convergence of the EM algorithm towards the maximum likelihood estimate (MLE) $\hat{\theta}$ depends on the smallest eigenvalue of

$$\mathcal{I}_C(\theta; y)^{-1} I_C(\theta; y).$$

The EM algorithm can converge very slowly, in particular, when the components of the mixture are not well separated, which indicates that these two matrices, $\mathcal{I}_C(\theta; y)$ and $I_C(\theta; y)$, can be quite different. Now the observed information matrix $I(\hat{\theta}; y)$ should be quite similar to the Fisher information matrix $F(\hat{\theta})$; and for component distributions belonging to the regular exponential family, $\mathcal{I}_C(\hat{\theta}; y)$ is equal to $F_C(\hat{\theta})$. This suggests that the determinants of $F_C(\hat{\theta})$ and $F(\hat{\theta})$ can be quite different.

Despite this there are situations in which the determinant of $F_C(\hat{\theta})$ may replace the determinant of $F(\hat{\theta})$ in (1) with little effect on the estimated parameters. Jorgensen [27] shows that, in the case of the single-factor analysis model studied by Wallace and Freeman [28], using an EM algorithm to implement a version of MML in which the determinant of $F(\hat{\theta})$ is replaced by that of $F_C\ (\hat{\theta})$ yields very similar results to those of [28]. Jorgensen [27] also shows that the evaluation of the determinant of $F(\hat{\theta})$ can be very intricate.

SNOB TODAY

Another strand in MML work in unsupervised learning began in 1998 with Wallace's article [29] in which Wallace considers strategies for incorporating spatial information into mixture model clustering. The basic setup is in terms of Markov random fields. A contribution to the research programme envisaged in [29] appears in [30] which gives more information about the message length approximations used.

The most recent reference on the Snob program as such is [31]. Snob has been extended to allow univariate Poisson and von Mises circular variables (attributes) in addition to the normal and discrete variables originally allowed. Because Snob assumes that all variables are independent within each cluster, it should not be difficult to extend Snob to cope with other types of variable distribution once the problem of MML estimation for such variables has been solved. There seem to be relatively few examples of successful MML approaches to genuine multivariate distributions, so extending Snob to have the ability to fit more complicated component distributions is a harder problem. However, Agusta and Dowe [32] have succeeded in developing an MML approach to fitting mixtures of multivariate normal distributions. It appears that Snob's model search strategy may have to be rethought as the possibility of groups of multivariate normal vectors of attributes opens up a much larger model space to be searched in.

Hunt and Jorgensen [33] consider the maximum likelihood fitting of mixture models similar to Snob for continuous and categorical variables. The continuous variables may be assumed to have block-diagonal covariance structure within mixture components. In the prostate cancer data studied by Hunt and Jorgensen [33], there was strong evidence of within-component associations in the two-, three-, and four-component mixtures fitted. However, the actual most probable assignments of the observations to clusters did not change markedly from the independent-within-clusters model when the stronger associations were added to the model, suggesting that Snob would have done well with this data. It is likely, though, that data sets exist in which allowing a more complex model in the components allows one to use substantially fewer components. Of course, it is just such trade-offs that MML is designed to evaluate, so perhaps one day there may be a Super-Snob with such capabilities.

ACKNOWLEDGEMENTS

The authors thank the late Professor Chris Wallace and Dr David Dowe for many helpful conversations about minimum message length inference and the Snob program.

REFERENCES

[1] Wallace, C.S. and Boulton, D.M. (1968) An information measure for classification. *Comput. J.*, **11**, 185–194.

[2] Boulton, D.M. and Wallace, C.S. (1970) A program for numerical classification. *Comput. J.*, **13**, 63–69.

[3] Wolfe, J. (1965) A Computer Program for the Computation of Maximum Likelihood Analysis of Types. Research Memo SRM 65-12. U.S. Naval Personnel Research Activity, San Diego.

[4] Wolfe, J. (1967) NORMIX: Computations for Estimating the Parameters of Multivariate Normal Mixtures of Distributions. Research Memo SRM 68-2. U.S. Naval Personnel Research Activity, San Diego.

[5] Wolfe, J.H. (1970) Pattern clustering by multivariate mixture analysis. *Multivariate Behav. Res.*, **5**, 329–350.

[6] Wolfe, J. (1971) A Monte Carlo Study of Sampling Distribution of the Likelihood Ratio for Mixtures Of Multi Normal Distributions. Technical Bulletin STB 672-2. U.S. Naval Personnel Research Activity, San Diego.

[7] Hasselblad, V. (1966) Estimation of parameters for a mixture of normal distributions. *Technometrics*, **8**, 431–444.

[8] Hasselblad, V. (1969) Estimation of finite mixtures of distributions from the exponential family. *J. Am. Statist. Assoc.*, **64**, 1459–1471.

[9] Day, N. (1969) Estimating the components of a mixture of two normal distributions. *Biometrika*, **56**, 463–474.

[10] Hartley, H. and Rao, J. (1968) Classification and estimation in analysis of variance problems. *Int. Statist. Rev.*, **36**, 141–147.

[11] Scott, A.J. and Symons, M.J. (1971) Clustering methods based on likelihood ratio criteria. *Biometries*, **27**, 387–397.

[12] Anderberg, M.R.C. (1973) *Cluster Analysis for Applications*. Academic Press, New York.

[13] Hartigan, J.A. (1975) *Clustering Algorithms*. Wiley, New York.

[14] Wallace, C.S. (1986) An Improved Program for Classification. *Proc. 9th Australian Computer Science Conf.*, pp. 357–366. Australian Computer Science Society.

[15] Dempster, A.P., Laird, N.M. and Rubin, D.B. (1977) Maximum likelihood from incomplete data via the EM algorithm (with discussion). *J. Roy. Statist. Soc. Ser. B*, **39**, 1–38.

[16] Banfield, J.D. and Raftery, A.E. (1993) Model-based Gaussian and non-Gaussian clustering. *Biometries*, **49**, 803–821.

[17] McLachlan, G.J. and Basford, K.E. (1988) *Mixture Models: Inference and Applications to Clustering*. Dekker, New York.

[18] Wallace, C.S. (2005) *Statistical and Inductive Inference by Minimum Message Length*. Information Science and Statistics. Springer, New York.

[19] McLachlan, G.J. and Krishnan, T. (1997) *The EM Algorithm and Extensions*. Wiley, New York.

[20] Jorgensen, M.A. (2001) EM algorithm. In El-Shaarawi, A.H. and Piegorsch, W.W.(eds), *Encyclopedia of Environmetrics*, Vol. 2, pp. 637–653. Wiley, New York.

[21] Wallace, C.S. (1984) An Improved Program for Classification. Technical Report 47. Department of Computer Science, Monash University, Melbourne.

[22] McLachlan, G.J. and Peel, D. (2000) *Finite Mixture Models*. Wiley, New York.

[23] Wallace, C.S. and Freeman, P.R. (1987) Estimation and inference by compact coding. *J. Roy. Statist. Soc. Ser. B*, **49**, 240–252.

[24] Conway, J.H. and Sloane, N.J.A. (1988) *Sphere Packings, Lattices and Groups*. Springer, London.

[25] Bouguila, N. and Ziou, D. (2007) High-dimensional unsupervised selection and estimation of a finite generalized Dirichlet mixture model based on minimum message length. *IEEE Trans. Pattern Anal. Mach. Intell.*, **29**, 1716–1731.

[26] Baxter, R.A. and Oliver, J.J. (2000) Finding overlapping components with MML. *Stat. Comput.*, **10**, 5–16.

[27] Jorgensen, M.A. (2005) Minimum message length estimation using EM methods: a case study. *Comput. Stat. Data Anal.*, **49**, 147–167.

[28] Wallace, C.S. and Freeman, P.R. (1992) Single factor analysis by minimum message length estimation. *J. Roy. Statist. Soc. Ser. B*, **54**, 195–209.

[29] Wallace, C.S. (1998) Intrinsic classification of spatially correlated data. *Comput. J.*, **41**, 602–411.

[30] Visser, G. and Dowe, D.L. (2007) Minimum Message Length Clustering of Spatially-Correlated Data with Varying Inter-Class Penalties. *Proc. 6th IEEE Int. Conf. on Computer and Information Science (ICIS)* y2007, 17–22.

[31] Wallace, C.S. and Dowe, D.L. (2000) MML mixture modelling of multi-state, Poisson, von Mises circular and Gaussian distributions. *Stat. Comput.*, **10**, 73–83.

[32] Agusta, Y. and Dowe, D.L. (2003) Unsupervised Learning of Correlated Multivariate Gaussian Mixture Models Using MML. In Gedeon, T.D. and Fung, L.C.C.(eds), *Lecture Notes in Computer Science*, Vol. 2903, pp. 477–489. Springer, New York.

[33] Hunt, L.A. and Jorgensen, M.A. (1999) Mixture model clustering using the *Multimix* program. *Aust. N. Z. J. Statist.*, **41**, 901–919.