MDL Histogram Density Estimation

Petri Kontkanen, Petri Myllymäki
Complex Systems Computation Group (CoSCo)
Helsinki Institute for Information Technology (HIIT)
University of Helsinki and Helsinki University of Technology
P.O.Box 68 (Department of Computer Science)
FIN-00014 University of Helsinki, Finland
{Firstname}.{Lastname}@hiit.fi

Abstract

We regard histogram density estimation as a model selection problem. Our approach is based on the information-theoretic minimum description length (MDL) principle, which can be applied for tasks such as data clustering, density estimation, image denoising and model selection in general. MDL-based model selection is formalized via the normalized maximum likelihood (NML) distribution, which has several desirable optimality properties. We show how this framework can be applied for learning generic, irregular (variable-width bin) histograms, and how to compute the NML model selection criterion efficiently. We also derive a dynamic programming algorithm for finding both the MDL-optimal bin count and the cut point locations in polynomial time. Finally, we demonstrate our approach via simulation tests.

1 INTRODUCTION

Density estimation is one of the central problems in statistical inference and machine learning. Given a sample of observations, the goal of *histogram density estimation* is to find a piecewise constant density that describes the data best according to some predetermined criterion. Although histograms are conceptually simple densities, they are very flexible and can model complex properties like multi-modality with a relatively small number of parameters. Furthermore, one does not need to assume any specific form for the underlying density function: given enough bins, a histogram estimator adapts to any kind of density.

Most existing methods for learning histogram densities assume that the bin widths are equal and concentrate only on finding the optimal bin count. These *regular* histograms are, however, often problematic. It has been argued (Rissanen, Speed, & Yu, 1992) that regular histograms are only good for describing roughly uniform data. If the data distribution is strongly non-uniform, the bin count must necessarily be high if one wants to capture the details of the high density portion of the data. This in turn means that an unnecessary large amount of bins is wasted in the low density region.

To avoid the problems of regular histograms one must allow the bins to be of variable width. For these *irregular* histograms, it is necessary to find the optimal set of *cut points* in addition to the number of bins, which naturally makes the learning problem essentially more difficult. For solving this problem, we regard the histogram density estimation as a model selection task, where the cut point sets are considered as models. In this framework, one must first choose a set of candidate cut points, from which the optimal model is searched for. The quality of each of the cut point sets is then measured by some model selection criterion.

Our approach is based on information theory, more specifically on the *Minimum description length* (MDL) principle developed in the series of papers (Rissanen, 1978, 1987, 1996). MDL is a well-founded, general framework for performing model selection and other types of statistical inference. The fundamental idea behind the MDL principle is that any regularity in data can be used to *compress* the data, i.e., to find a description or *code* of it such that this description uses the least number of symbols, less than other codes and less than it takes to describe the data literally. The more regularities there are, the more the data can be compressed. According to the MDL principle, learning can be equated with finding regularities in data. Consequently, we can say that the more we are able to compress the data, the more we have learned about it.

Model selection with MDL is done by minimizing a

quantity called *the stochastic complexity*, which is the shortest description length of a given data relative to a given model class. The definition of the stochastic complexity is based on the *normalized maximum likelihood* (NML) distribution introduced in (Shtarkov, 1987; Rissanen, 1996). The NML distribution has several theoretical optimality properties, which make it a very attractive candidate for performing model selection. It was originally (Rissanen, 1996) formulated as a unique solution to the minimax problem presented in (Shtarkov, 1987), which implied that NML is the minimax optimal universal model. Later (Rissanen, 2001), it was shown that NML is also the solution to a related problem involving expected regret. See Section 2 and (Rissanen, 2001; Grünwald, 2006; Rissanen, 2005) for more discussion on the theoretical properties of the NML.

On the practical side, NML has been successfully applied to several problems. We mention here two examples. In (Kontkanen, Myllymäki, Buntine, Rissanen, & Tirri, 2006), NML was used for data clustering, and its performance was compared to alternative approaches like Bayesian statistics. The results showed that NML was especially impressive with small sample sizes. In (Roos, Myllymäki, & Tirri, 2005), NML was applied to wavelet denoising of computer images. Since the MDL principle in general can be interpreted as separating information from noise, this approach is very natural.

Unfortunately, in most practical applications of NML one must face severe computational problems, since the definition of the NML involves a normalizing integral or a sum, called the *parametric complexity*, which usually is difficult to compute. One of the contributions of this paper is to show how the parametric complexity can be computed efficiently in the histogram case, which makes it possible to use NML as a model selection criterion in practice.

There is obviously an exponential number of different cut point sets. Therefore, a brute-force search is not feasible. Another contribution of this paper is to show how the NML-optimal cut point locations can be found via dynamic programming in a polynomial (quadratic) time with respect to the size of the set containing the cut points considered in the optimization process.

The histogram density estimation is naturally a well-studied problem, but unfortunately almost all of the previous studies, e.g. (Birge & Rozenholc, 2002; Hall & Hannan, 1988; Yu & Speed, 1992), consider regular histograms only. Most similar to our work is (Rissanen et al., 1992), in which irregular histograms are learned with the Bayesian mixture criterion using a uniform prior. The same criterion is also used in (Hall & Hannan, 1988), but the histograms are equal-width only. Another similarity between our work and (Rissanen et al., 1992) is the dynamic programming optimization process, but since the optimality criterion is not the same, the process itself is quite different. It should be noted that these differences are significant as the Bayesian mixture criterion does not possess the optimality properties of NML mentioned above.

This paper is structured as follows. In Section 2 we discuss the basic properties of the MDL framework in general, and also shortly review the optimality properties of the NML distribution. Section 3 introduces the NML histogram density and also provides a solution to the related computational problem. The cut point optimization process based on dynamic programming is the topic of Section 4. Finally, in Section 5 our approach is demonstrated via simulation tests.

2 PROPERTIES OF MDL AND NML

The MDL principle has several desirable properties. Firstly, it automatically protects against overfitting when learning both the parameters and the structure (number of parameters) of the model. Secondly, there is no need to assume the existence of some underlying "true" model, which is not the case with several other statistical methods. The model is only used as a technical device for constructing an efficient code. MDL is also closely related to the Bayesian inference but there are some fundamental differences, the most important being that MDL is not dependent on any prior distribution, it only uses the data at hand.

MDL model selection is based on minimization of the stochastic complexity. In the following, we give the definition of the stochastic complexity and then proceed by discussing its theoretical properties.

Let $\mathbf{x}^n = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ be a data sample of n outcomes, where each outcome $\mathbf{x}_j$ is an element of some space of observations $\mathcal{X}$. The n-fold cartesian product $\mathcal{X} \times \cdots \times \mathcal{X}$ is denoted by $\mathcal{X}^n$, so that $\mathbf{x}^n \in \mathcal{X}^n$. Consider a set $\Theta \subseteq \mathbb{R}^d$, where d is a positive integer. A class of parametric distributions indexed by the elements of Θ is called a *model class*. That is, a model class $\mathcal{M}$ is defined as $\mathcal{M} = \{f(\cdot \mid \theta) : \theta \in \Theta\}$. Denote the maximum likelihood estimate of data $\mathbf{x}^n$ by $\hat{\theta}(\mathbf{x}^n)$, i.e.,

$$\hat{\theta}(\mathbf{x}^n) = \arg\max_{\theta \in \Theta} \{f(\mathbf{x}^n \mid \theta)\}. \tag{1}$$

The *normalized maximum likelihood (NML)* density (Shtarkov, 1987) is now defined as

$$f_{\mathrm{NML}}(\mathbf{x}^n \mid \mathcal{M}) = \frac{f(\mathbf{x}^n \mid \hat{\theta}(\mathbf{x}^n), \mathcal{M})}{\mathcal{R}^n_{\mathcal{M}}}, \tag{2}$$

where the normalizing constant $\mathcal{R}^n_{\mathcal{M}}$ is given by

$$\mathcal{R}^n_{\mathcal{M}} = \int_{\mathbf{x}^n \in \mathcal{X}^n} f(\mathbf{x}^n \mid \hat{\theta}(\mathbf{x}^n), \mathcal{M}) d\mathbf{x}^n, \quad (3)$$

and the range of integration goes over the space of data samples of size n. If the data is discrete, the integral is replaced by the corresponding sum.

The stochastic complexity of the data $\mathbf{x}^n$ given a model class $\mathcal{M}$ is defined via the NML density as

$$SC(\mathbf{x}^n \mid \mathcal{M}) = -\log f_{\mathrm{NML}}(\mathbf{x}^n \mid \mathcal{M}) \quad (4)$$

$$= -\log f(\mathbf{x}^n \mid \hat{\theta}(\mathbf{x}^n), \mathcal{M}) + \log \mathcal{R}^n_{\mathcal{M}}, \quad (5)$$

and the term $\log \mathcal{R}^n_{\mathcal{M}}$ is called the *parametric complexity* or *minimax regret*. The parametric complexity can be interpreted as measuring the logarithm of the number of essentially different (distinguishable) distributions in the model class. Intuitively, if two distributions assign high likelihood to the same data samples, they do not contribute much to the overall complexity of the model class, and the distributions should not be counted as different for the purposes of statistical inference. See (Balasubramanian, 2006) for more discussion on this topic.

The NML density (2) has several important theoretical optimality properties. The first one is that NML provides a unique solution to the minimax problem posed in (Shtarkov, 1987),

$$\min_{\hat{f}} \max_{\mathbf{x}^n} \log \frac{f(\mathbf{x}^n \mid \hat{\theta}(\mathbf{x}^n), \mathcal{M})}{\hat{f}(\mathbf{x}^n \mid \mathcal{M})} = \log \mathcal{R}^n_{\mathcal{M}}, \quad (6)$$

This means that the NML density is the *minimax optimal universal model*. A related property of NML involving expected regret was proven in (Rissanen, 2001). This property states that NML also minimizes

$$\min_{\hat{f}} \max_{g} E_g \log \frac{f(\mathbf{x}^n \mid \hat{\theta}(\mathbf{x}^n), \mathcal{M})}{\hat{f}(\mathbf{x}^n \mid \mathcal{M})} = \log \mathcal{R}^n_{\mathcal{M}}, \quad (7)$$

where the expectation is taken over $\mathbf{x}^n$ and g is the worst-case data generating density.

Having now discussed the MDL principle and the NML density in general, we return to the main topic of the paper. In the next section, we instantiate the NML density for the histograms and show how the parametric complexity can be computed efficiently in this case.

3 NML HISTOGRAM DENSITY

Consider a sample of n outcomes $\mathbf{x}^n = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$ on the interval $[\mathbf{x}_{\min}, \mathbf{x}_{\max}]$. Typically, $\mathbf{x}_{\min}$ and $\mathbf{x}_{\max}$ are defined as the minimum and maximum value in $\mathbf{x}^n$, respectively. Without any loss of generality, we assume that the data is sorted into increasing order. Furthermore, we assume that the data is recorded at a finite accuracy ϵ, which means that each $\mathbf{x}_j \in \mathbf{x}^n$ belongs to the set $\mathcal{X}$ defined by

$$\mathcal{X} = \{\mathbf{x}_{\min} + t\epsilon : t = 0, \ldots, \frac{\mathbf{x}_{\max} - \mathbf{x}_{\min}}{\epsilon}\}. \quad (8)$$

This assumption is made to simplify the mathematical formulation, and as can be seen later, the effect of the accuracy parameter ϵ on the stochastic complexity is a constant that can be ignored in the model selection process.

Let $C = (c_1, \ldots, c_{K-1})$ be an increasing sequence of points partitioning the range $[\mathbf{x}_{\min} - \epsilon/2, \mathbf{x}_{\max} + \epsilon/2]$ into the following K intervals (bins):

$$([\mathbf{x}_{\min} - \epsilon/2, c_1],]c_1, c_2], \ldots,]c_{K-1}, \mathbf{x}_{\max} + \epsilon/2]). \quad (9)$$

The points c_k are called the *cut points* of the histogram. Note that the original data range $[\mathbf{x}_{\min}, \mathbf{x}_{\max}]$ is extended by $\epsilon/2$ from both ends for technical reasons. It is natural to assume that there is only one cut point between two consecutive elements of $\mathcal{X}$, since placing two or more cut points would always produce unnecessary empty bins. For simplicity, we assume that the cut points belong to the set $\mathcal{C}$ defined by

$$\mathcal{C} = \{\mathbf{x}_{\min} + \epsilon/2 + t\epsilon : t = 0, \ldots, \frac{\mathbf{x}_{\max} - \mathbf{x}_{\min}}{\epsilon} - 1\}, \quad (10)$$

i.e., each $c_k \in \mathcal{C}$ is a midpoint of two consecutive values of $\mathcal{X}$.

Define $c_0 = \mathbf{x}_{\min} - \epsilon/2, c_K = \mathbf{x}_{\max} + \epsilon/2$ and let $L_k = c_k - c_{k-1}, k = 1, \ldots, K$ be the bin lengths. Given a parameter vector $\theta \in \Theta$,

$$\Theta = \{(\theta_1, \ldots, \theta_K) : \theta_k \geq 0, \theta_1 + \cdots + \theta_K = 1\}, \quad (11)$$

and a set (sequence) of cut points C, we now define the histogram density f_h by

$$f_h(x \mid \theta, C) = \frac{\epsilon \cdot \theta_k}{L_k}, \quad (12)$$

where $x \in]c_{k-1}, c_k]$. Note that (12) does not define a density in the purest sense, since $f_h(x \mid \theta, C)$ is actually the probability that x falls into the interval $]x - \epsilon/2, x + \epsilon/2]$. Given (12), the likelihood of the whole data sample $\mathbf{x}^n$ is easy to write. We have

$$f_h(\mathbf{x}^n \mid \theta, C) = \prod_{k=1}^{K} \left(\frac{\epsilon \cdot \theta_k}{L_k} \right)^{h_k}, \quad (13)$$

where h_k is the number of data points falling into bin k.

To instantiate the NML distribution (2) for the histogram density f_h, we need to find the maximum likelihood parameters $\hat{\theta}(x^n) = (\hat{\theta}_1, \ldots, \hat{\theta}_K)$ and an efficient way to compute the parametric complexity (3). It is well-known that the ML parameters are given by the relative frequencies $\hat{\theta}_k = h_k/n$, so that we have

$$f_h(\mathbf{x}^n \mid \hat{\theta}(\mathbf{x}^n), C) = \prod_{k=1}^{K} \left(\frac{\epsilon \cdot h_k}{L_k \cdot n} \right)^{h_k}. \tag{14}$$

Denote now the parametric complexity of a K-bin histogram by $\log \mathcal{R}^n_{h_K}$. First thing to notice is that since the data is pre-discretized, the integral in (3) is replaced by a sum over the space $\mathcal{X}^n$. We have

$$\begin{aligned}
\mathcal{R}^n_{h_K} &= \sum_{\mathbf{x}^n \in \mathcal{X}^n} \prod_{k=1}^{K} \left(\frac{\epsilon \cdot h_k}{L_k \cdot n} \right)^{h_k} && (15)\\
&= \sum_{h_1 + \cdots + h_K = n} \frac{n!}{h_1! \cdots h_K!} \prod_{k=1}^{K} \left(\frac{L_k}{\epsilon} \right)^{h_k} \\
&\qquad \cdot \prod_{k=1}^{K} \left(\frac{\epsilon \cdot h_k}{L_k \cdot n} \right)^{h_k} && (16)\\
&= \sum_{h_1 + \cdots + h_K = n} \frac{n!}{h_1! \cdots h_K!} \prod_{k=1}^{K} \left(\frac{h_k}{n} \right)^{h_k}, && (17)
\end{aligned}$$

where the term $(L_k/\epsilon)^{h_k}$ in (16) follows from the fact that an interval of length L_k contains exactly (L_k/ϵ) members of the set $\mathcal{X}$, and the multinomial coefficient $n!/(h_1! \cdots h_K!)$ counts the number of arrangements of n objects into K boxes each containing $h_1, \ldots, h_K$ objects, respectively.

Although the final form (17) of the parametric complexity is still an exponential sum, we can compute it efficiently. It turns out that (17) is exactly the same as the parametric complexity of a K-valued multinomial, which we studied in (Kontkanen & Myllymäki, 2005). In this work, we derived the recursion

$$\mathcal{R}^n_{h_K} = \mathcal{R}^n_{h_{K-1}} + \frac{n}{K-2} \mathcal{R}^n_{h_{K-2}}, \tag{18}$$

which holds for $K > 2$. It is now straightforward to write a linear-time algorithm based on (18). The computation starts with the trivial case $\mathcal{R}^n_{h_1} \equiv 1$. The case $K = 2$ is a simple sum

$$\mathcal{R}^n_{h_2} = \sum_{h_1 + h_2 = n} \frac{n!}{h_1! h_2!} \left(\frac{h_1}{n} \right)^{h_1} \left(\frac{h_2}{n} \right)^{h_2}, \tag{19}$$

which clearly can be computed in time $\mathcal{O}(n)$. Finally, recursion (18) is applied $K - 2$ times to end up with $\mathcal{R}^n_{h_K}$. The time complexity of the whole computation is $\mathcal{O}(n + K)$.

Having now derived both the maximum likelihood parameters and the parametric complexity, we are now ready to write down the stochastic complexity (5) for the histogram model. We have

$$\begin{aligned}
&SC(\mathbf{x}^n \mid C) \\
&\quad = -\log \frac{\prod_{k=1}^{K} \left(\frac{\epsilon \cdot h_k}{L_k \cdot n} \right)^{h_k}}{\mathcal{R}^n_{h_K}} && (20)\\
&\quad = \sum_{k=1}^{K} -h_k (\log(\epsilon \cdot h_k) - \log(L_k \cdot n)) \\
&\qquad + \log \mathcal{R}^n_{h_K}. && (21)
\end{aligned}$$

Equation (21) is the basis for measuring the quality of NML histograms, i.e., comparing different cut point sets. It should be noted that as the term $\sum_{k=1}^{K} -h_k \log \epsilon = -n \log \epsilon$ is a constant with respect to C, the value of ϵ does not affect the comparison. In the next section we will discuss how NML-optimal histograms can be found in practice.

4 LEARNING MDL-OPTIMAL HISTOGRAMS

In this section we will describe a dynamic programming algorithm, which can be used to efficiently find both the optimal bin count and the cut point locations. We start by giving the exact definition of the problem. Let $\tilde{\mathcal{C}} \subseteq \mathcal{C}$ denote the *candidate cut point set*, which is the set of cut points we consider in the optimization process. How $\tilde{\mathcal{C}}$ is chosen in practice, depends on the problem at hand. The simplest choice is naturally $\tilde{\mathcal{C}} = \mathcal{C}$, which means that all the possible cut points are candidates. However, if the value of the accuracy parameter ϵ is small or the data range contains large gaps, this choice might not be practical. Another idea would be to define $\tilde{\mathcal{C}}$ to be the set of midpoints of all the consecutive value pairs in the data $\mathbf{x}^n$. This choice, however, does not allow empty bins, and thus the potential large gaps are still problematic.

A much more sensible choice is to place two candidate cut points between each consecutive values in the data. It is straightforward to prove and also intuitively clear that these two candidate points should be placed as close as possible to the respective data points. In this way, the resulting bin lengths are as small as possible, which will produce the greatest likelihood for the data. These considerations suggest that $\tilde{\mathcal{C}}$ should be chosen as

$$\begin{aligned}
\tilde{\mathcal{C}} =& (\{\mathbf{x}_j - \epsilon/2 : \mathbf{x}_j \in \mathbf{x}^n\} \cup \{\mathbf{x}_j + \epsilon/2 : \mathbf{x}_j \in \mathbf{x}^n\}) \\
& \setminus \{\mathbf{x}_{\min} - \epsilon/2, \mathbf{x}_{\max} + \epsilon/2\}. && (22)
\end{aligned}$$

Note that the end points $\mathbf{x}_{\min} - \epsilon/2$ and $\mathbf{x}_{\max} + \epsilon/2$

are excluded from $\tilde{\mathcal{C}}$, since they are always implicitly included in all the cut point sets.

After choosing the candidate cut point set, the histogram density estimation problem is straightforward to define: find the cut point set $C \subseteq \tilde{\mathcal{C}}$ which optimizes the given goodness criterion. In our case the criterion is based on the stochastic complexity (21), and the cut point sets are considered as models. In practical model selection tasks, however, the stochastic complexity criterion itself may not be sufficient. The reason is that it is also necessary to encode the model index in some way, as argued in (Grünwald, 2006). In some tasks, an encoding based on the uniform distribution is appropriate. Typically, if the set of models is finite and the models are of same complexity, this choice is suitable. In the histogram case, however, the cut point sets of different size produce densities which are dramatically different complexity-wise. Therefore, it is natural to assume that the model index is encoded with a uniform distribution over all the cut point sets of the same size. For a K-bin histogram with the size of the candidate cut point set fixed to E, there are clearly $\binom{E}{K-1}$ ways to choose the cut points. Thus, the codelength for encoding them is $\log \binom{E}{K-1}$.

After these considerations, we define the final criterion (or score) used for comparing different cut point sets as

$$\begin{aligned} &B(\mathbf{x}^n \mid E, K, C) \\ &\quad = SC(\mathbf{x}^n \mid C) + \log \binom{E}{K-1} \qquad (23) \\ &\quad = \sum_{k=1}^{K} -h_k \left(\log(\epsilon \cdot h_k) - \log(L_k \cdot n)\right) \\ &\qquad + \log \mathcal{R}_{h_K}^n + \log \binom{E}{K-1}. \qquad (24) \end{aligned}$$

It is clear that there is an exponential number of possible cut point sets, and thus an exhaustive search to minimize (24) is not feasible. However, the optimal cut point set can be found via dynamic programming, which works by tabulating partial solutions to the problem. The final solution is then found recursively.

Let us first assume that the elements of $\tilde{\mathcal{C}}$ are indexed in such a way that

$$\tilde{\mathcal{C}} = \{\tilde{c}_1, \ldots, \tilde{c}_E\}, \ \tilde{c}_1 < \tilde{c}_2 < \cdots < \tilde{c}_E. \qquad (25)$$

We also define $\tilde{c}_{E+1} = \mathbf{x}_{\max} + \epsilon/2$. Denote

$$\hat{B}_{K,e} = \min_{C \subseteq \tilde{\mathcal{C}}} B(\mathbf{x}^{n_e} \mid E, K, C), \qquad (26)$$

where $\mathbf{x}^{n_e} = (\mathbf{x}_1, \ldots, \mathbf{x}_{n_e})$ is the portion of the data falling into interval $[\mathbf{x}_{\min}, \tilde{c}_e]$ for $e = 1, \ldots, E+1$. This means that $\hat{B}_{K,e}$ is the optimizing value of (24) when the data is restricted to $\mathbf{x}^{n_e}$. For a fixed K, $\hat{B}_{K,E+1}$ is clearly the final solution we are looking for, since the interval $[\mathbf{x}_{\min}, \tilde{c}_{E+1}]$ contains all the data.

Consider now a K-bin histogram with cut points $C = (\tilde{c}_{e_1}, \ldots, \tilde{c}_{e_{K-1}})$. Assuming that the data range is restricted to $[\mathbf{x}_{\min}, \tilde{c}_{e_K}]$ for some $\tilde{c}_{e_K} > \tilde{c}_{e_{K-1}}$, we can straightforwardly write the score function $B(\mathbf{x}^{n_{e_K}} \mid E, K, C)$ by using the score function of a $(K-1)$-bin histogram with cut points $C' = (\tilde{c}_{e_1}, \ldots, \tilde{c}_{e_{K-2}})$ as

$$\begin{aligned} &B(\mathbf{x}^{n_{e_K}} \mid E, K, C) \\ &\quad = B(\mathbf{x}^{n_{e_{K-1}}} \mid E, K-1, C') \\ &\qquad - (n_{e_K} - n_{e_{K-1}})(\log(\epsilon \cdot (n_{e_K} - n_{e_{K-1}})) \\ &\qquad - \log((\tilde{c}_{e_K} - \tilde{c}_{e_{K-1}}) \cdot n)) \\ &\qquad + \log \frac{\mathcal{R}_{h_K}^{n_{e_K}}}{\mathcal{R}_{h_{K-1}}^{n_{e_{K-1}}}} + \log \frac{E-K+2}{K-1}, \qquad (27) \end{aligned}$$

since $(n_{e_K} - n_{e_{K-1}})$ is the number of data points falling into the K^{th} bin, $(\tilde{c}_{e_K} - \tilde{c}_{e_{K-1}})$ is the length of that bin, and

$$\log \frac{\binom{E}{K-1}}{\binom{E}{K-2}} = \log \frac{E-K+2}{K-1}. \qquad (28)$$

We can now write the dynamic programming recursion as

$$\begin{aligned} \hat{B}_{K,e} = \min_{e'} \Big\{ &\hat{B}_{K-1,e'} - (n_e - n_{e'}) \cdot (\log(\epsilon \cdot (n_e - n_{e'})) \\ &- \log((\tilde{c}_e - \tilde{c}_{e'}) \cdot n)) \\ &+ \log \frac{\mathcal{R}_{h_K}^{n_e}}{\mathcal{R}_{h_{K-1}}^{n_{e'}}} + \log \frac{E-K+2}{K-1} \Big\}, \qquad (29) \end{aligned}$$

where $e' = K-1, \ldots, e-1$. The recursion is initialized with

$$\hat{B}_{1,e} = -n_e \cdot (\log(\epsilon \cdot n_e) - \log((\tilde{c}_e - (\mathbf{x}_{\min} - \epsilon/2)) \cdot n)), \qquad (30)$$

for $e = 1, \ldots, E+1$. After that, the bin count is always increased by one, and (29) is applied for $e = K, \ldots, E+1$ until a pre-determined maximum bin count $K_{\max}$ is reached. The minimum $\hat{B}_{K,e}$ is then chosen to be the final solution. By constantly keeping track which e' minimizes (29) during the process, the optimal cut point sequence can also be recovered. The time complexity of the whole algorithm is $\mathcal{O}\left(E^2 \cdot K_{\max}\right)$.

5 EMPIRICAL RESULTS

The quality of a density estimator is usually measured by a suitable distance metric between the data generating density and the estimated one. This is often

problematic, since we typically do not know the data generating density, which means that some heavy assumptions must be made. The MDL principle, however, states that the stochastic complexity (plus the codelength for encoding the model index) itself can be used as a goodness measure. Therefore, it is not necessary to use any additional way of assessing the quality of an MDL density estimator. The optimality properties of the NML criterion and the fact that we are able to find the global optimum in the histogram case will make sure that the final result is theoretically valid.

Nevertheless, to demonstrate the behaviour of the NML histogram method in practice we implemented the dynamic programming algorithm of the previous section and ran some simulation tests. We generated data samples of various size from four densities of different shapes (see below) and then used the dynamic programming method to find the NML-optimal histograms. In all the tests, the accuracy parameter ϵ was fixed to 0.1. We decided to use Gaussian finite mixtures as generating densities, since they are very flexible and easy to sample from. The four generating densities we chose and the corresponding NML-optimal histograms using a sample of 10000 data points are shown in Figures 1 and 2. The densities are labeled gm2, gm5, gm6 and gm8, and they are mixtures of 2, 5, 6 and 8 Gaussian components, respectively, with various amount of overlap between the components. From the plots we can see that the NML histogram method is able to capture properties such as multi-modality (all densities) and long tails (gm6). Another nice feature is that the algorithm automatically places more bins to the areas where more detail is needed like the high, narrow peaks of gm5 and gm6.

To see the behaviour of the NML histogram density algorithm with varying amount of data, we generated data samples of various sizes between $100-10000$ from the four generating densities. For each case, we measured the distance between the generating density and the NML-optimal histogram. As the distance measure we used the (squared) Hellinger distance

$$h^2(f,g) = \int (\sqrt{f(x)} - \sqrt{g(x)})^2 dx, \qquad (31)$$

which has often been used in the histogram context before (see, e.g., (Birge & Rozenholc, 2002; Kanazawa, 1993)). The actual values of the Hellinger distance were calculated via numerical integration. The results can be found in Figure 3. The curves are averaged over 10 different samples of each size. The figure shows that the NML histogram density converges to the generating one quite rapidly when the sample size is increased. The shapes of the convergence curves with the four generating densities are also very similar, which is further evidence of the flexibility of the variable-width

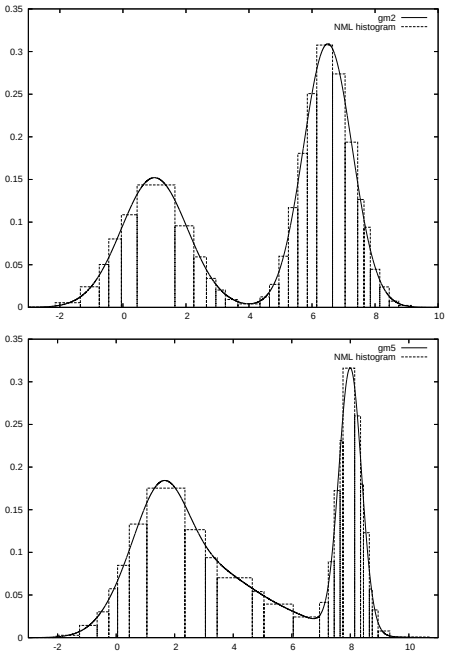

Figure 1: The Gaussian finite mixture densities gm2 and gm5 and the NML-optimal histograms with sample size 10000.

histograms.

To visually see the effect of the sample size, we plotted the NML-optimal histograms against the generating density gm6 with sample sizes 100, 1000 and 10000. These plots can be found in Figure 4. As a reference, we also plotted the empirical distributions of the data samples as a (mirrored) equal-width histograms (the negative y-values). Each bar of the empirical plot has width 0.1 (the value of the accuracy parameter ϵ). When the sample size is 100, the NML histogram algorithm has chosen only 3 bins, and the resulting histogram density is rather crude. However, the small sample size does not justify placing any more bins as can be seen from the empirical distribution. Therefore, we claim that the NML-optimal solution is actually a very sensible one. When the sample size is increased, the bin count is increased and more and more details are captured. Notice that with all the sample sizes, the bin widths of the NML-optimal histograms are strongly variable. It is clear that it would be impossible for any equal-width histogram density estimator to produce such detailed results using the same amount of data.

6 CONCLUSION

In this paper we have presented an information-theoretic framework for histogram density estimation.

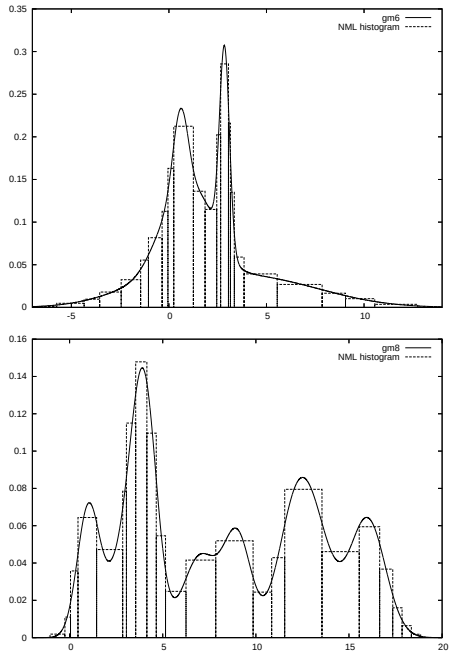

Figure 2: The Gaussian finite mixture densities gm6 and gm8 and the NML-optimal histograms with sample size 10000.

The selected approach based on the MDL principle has several advantages. Firstly, the MDL criterion for model selection (stochastic complexity) has nice theoretical optimality properties. Secondly, by regarding histogram estimation as a model selection problem, it is possible to learn generic, variable-width bin histograms and also estimate the optimal bin count automatically. Furthermore, the MDL criterion itself can be used as a measure of quality of a density estimator, which means that there is no need to assume anything about the underlying generating density. Since the model selection criterion is based on the NML distribution, there is also no need to specify any prior distribution for the parameters.

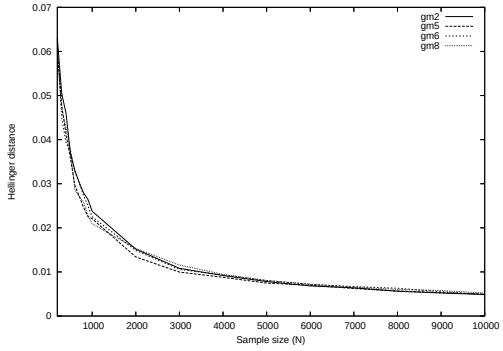

Figure 3: The Hellinger distance between the four generating densities and the corresponding NML-optimal histograms as a function of the sample size.

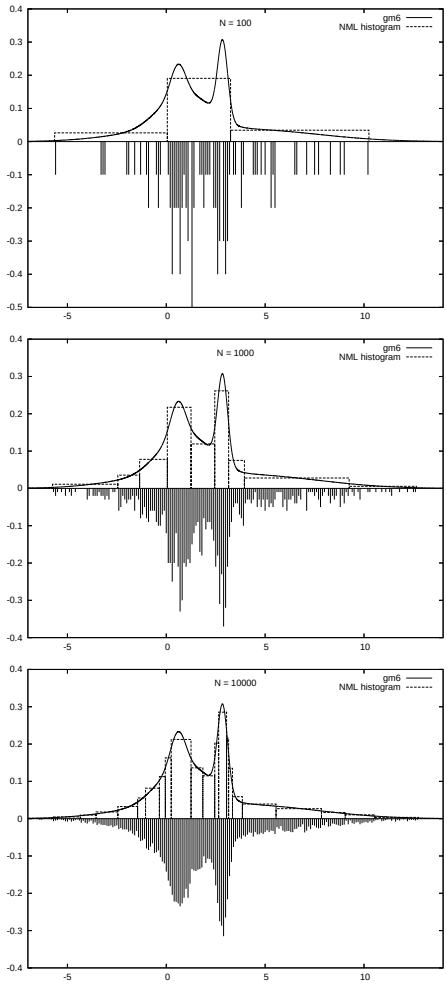

Figure 4: The generating density gm6, the NML-optimal histograms and the empirical distributions with sample sizes 100, 1000 and 10000.

To make our approach practical, we presented an efficient way to compute the value of the stochastic complexity in the histogram case. We also derived a dynamic programming algorithm for efficiently optimizing the NML-based criterion. Consequently, we were able to find the globally optimal bin count and cut point locations in quadratic time with respect to the size of the candidate cut point set.

In addition to the theoretical part, we demonstrated the validity of our approach by simulation tests. In these tests, data samples of various sizes were generated from Gaussian finite mixture densities with highly complex shapes. The results showed that the NML histograms automatically adapt to various kind of densities.

In the future, our plan is to perform an extensive set of empirical tests using both simulated and real data. In these tests, we will compare our approach to other histogram estimators. It is anticipated that the various equal-width estimators will not be performing well in the tests due to the severe limitations of regular histograms. More interesting will be the comparative performance of the density estimator in (Rissanen et al., 1992), which is similar to ours but based on the Bayesian mixture criterion. Theoretically, our version has an advantage at least with small sample sizes.

Another interesting application of NML histograms would be to use them for modeling the class-specific distributions of classifiers such as the Naive Bayes. These distributions are usually modeled with a Gaussian density or a multinomial distribution with equal-width discretization, which typically cannot capture all the relevant properties of the distributions. Although the NML histogram is not specifically tailored for classification tasks, it seems evident that if the class-specific distributions are modeled with high accuracy, the resulting classifier also performs well.

Acknowledgements

This work was supported in part by the Academy of Finland under the project Civi and by the Finnish Funding Agency for Technology and Innovation under the projects Kukot and PMMA. In addition, this work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

References

Balasubramanian, V. (2006). MDL, Bayesian inference, and the geometry of the space of probability distributions. In P. Grünwald, I. Myung, & M. Pitt (Eds.), *Advances in minimum description length: Theory and applications* (pp. 81–98). The MIT Press.

Birge, L., & Rozenholc, Y. (2002, April). *How many bins should be put in a regular histogram.* (Prepublication no 721, Laboratoire de Probabilites et Modeles Aleatoires, CNRS-UMR 7599, Universite Paris VI & VII)

Grünwald, P. (2006). Minimum description length tutorial. In P. Grünwald, I. Myung, & M. Pitt (Eds.), *Advances in minimum description length: Theory and applications* (pp. 23–79). The MIT Press.

Hall, P., & Hannan, E. (1988). On stochastic complexity and nonparametric density estimation. *Biometrika*, *75*(4), 705–714.

Kanazawa, Y. (1993). Hellinger distance and Akaike's information criterion for the histogram. *Statist. Probab. Letters*(17), 293–298.

Kontkanen, P., & Myllymäki, P. (2005). *Analyzing the stochastic complexity via tree polynomials* (Tech. Rep. No. 2005-4). Helsinki Institute for Information Technology (HIIT).

Kontkanen, P., Myllymäki, P., Buntine, W., Rissanen, J., & Tirri, H. (2006). An MDL framework for data clustering. In P. Grünwald, I. Myung, & M. Pitt (Eds.), *Advances in minimum description length: Theory and applications.* The MIT Press.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*, 445-471.

Rissanen, J. (1987). Stochastic complexity. *Journal of the Royal Statistical Society*, *49*(3), 223–239 and 252–265.

Rissanen, J. (1996). Fisher information and stochastic complexity. *IEEE Transactions on Information Theory*, *42*(1), 40–47.

Rissanen, J. (2001). Strong optimality of the normalized ML models as universal codes and information in data. *IEEE Transactions on Information Theory*, *47*(5), 1712–1717.

Rissanen, J. (2005, August). *Lectures on statistical modeling theory.* (Available online at www.mdl-research.org)

Rissanen, J., Speed, T., & Yu, B. (1992). Density estimation by stochastic complexity. *IEEE Transactions on Information Theory*, *38*(2), 315–323.

Roos, T., Myllymäki, P., & Tirri, H. (2005). On the behavior of MDL denoising. In R. G. Cowell & Z. Ghahramani (Eds.), *Proceedings of the 10th international workshop on artificial intelligence and statistics (aistats)* (pp. 309–316). Barbados: Society for Artificial Intelligence and Statistics.

Shtarkov, Y. M. (1987). Universal sequential coding of single messages. *Problems of Information Transmission*, *23*, 3–17.

Yu, B., & Speed, T. (1992). Data compression and histograms. *Probab. Theory Relat. Fields*, *92*, 195–229.