An Empirical Investigation of the MDL Principle

Tim Van Allen **Chris Dutchyn** **Russ Greiner**
Department of Computing Science
Edmonton, AB T6G 2H1 Canada
{ vanallen, dutchyn, greiner }@cs.ualberta.ca

Abstract

This paper provides an empirical exploration of the "minimum description length" (MDL) principle, in the context of learning Bayesian belief nets (BNs). In one set of experiments, with relatively few variables, we comprehensively constructed the entire set of BN-structures, while in other tests, dealing with larger sets of variables, we carefully subsampled the space of structures. In each situation, we compared the BN with the smallest MDL score to various other BNs, including the "fully independent", "complete", and Chow Liu networks, to see which had the best "true likelihood" score, over the entire distribution of tuples. Our findings partially characterize when MDL is an appropriate heuristic, and when it is not.

Keywords: MDL, Learning, (Bayesian) Belief Nets, Overfitting, Foundations

1 Introduction

Bayesian belief nets (BNs) [Pea 88] provide a succinct way to encode a general joint distribution over a set of probabilistic variables, and they have proven to be an effective tool for many tasks. To help produce these needed BNs, there are now many algorithms for learning them from a set of training data. Most attempt to find the BN that is the "closest fit" to the underlying distribution; typically, which has the smallest Kullback-Liebler divergence. An obvious approach is to seek the BN that has the smallest divergence *from the training sample*. Unfortunately, this can lead to *overfitting* — as the BN with the smallest divergence from the training sample is not necessarily the one with the smallest divergence from the true distribution. This is especially true as this empirical divergence measure can only decrease as we expand the size of the BN.

We therefore need some "regularizing" term, to prevent our learners from simply selecting the largest possible network structure.[1] The "minimum description length" (MDL) maxim serves this function. It says, in effect, we should prefer a smaller belief net over a larger one, unless the larger one has a *significantly* better fit to the data — enough to justify the increase in size. (We provide the details in Section 2.)

Many learning algorithms use an MDL criterion when deciding between different BNs; many of the approaches appear to work effectively. Moreover, MDL is provably optimal *in the limit*. However, MDL is still only a heuristic — one which does not have to work, in the real situation where there is only a limited sample of data.

This report provides an *empirical* investigation of this principle. The next section provides the framework, overviewing related work, belief nets and their associated learning algorithms, and the MDL principle. Section 3 then presents our experimental framework, describing how we explore the space of network structures and how we evaluate the quality of the resulting BNs. Section 4 presents our results, together with our description of situations where MDL appears to work most effectively, and where it does not. We conclude with a discussion of additional areas which can extend this investigation.

2 Background

Occam's razor is an inductive bias that prefers "simpler" hypotheses (the notion of complexity is problematic) over more complex ones. Blumer *et al.* [Blu 86] formalized the notion of an "Occam algorithm", within

[1] As another reason to avoid huge structures, note that they may exhaust available storage resources or fail to give speedy responses to subsequent queries.

the PAC-learning framework [Val 84]. The basic idea is that, if you consider only short hypotheses, then only a small (polynomial) number of samples is necessary to reject all unacceptable hypotheses; hence, if this hypothesis space contains an acceptable hypothesis, it can be found in a reasonable amount of time. This follows simply from the observation that there are relatively few short hypotheses. However, the same result holds for *any* small set of hypotheses, regardless of their complexity under some arbitrary encoding scheme. Conversely, any small set has a simple (short) encoding scheme over its elements, even if their individual complexity (for example, in conjunctive normal form logic) is exponentially larger. Of course, Occam's razor is merely a bias, which does not have to work everywhere. This has prompted others to empirically investigate its effectiveness; in particular, Murphy and Pazzani [Mur 95] carry out an empirical investigation of Occam's razor (in the context of learning decision trees) that is similar in spirit to our current investigation of the MDL principle.

Rissanen [Ris 85] introduced the **minimum description length** (MDL) principle. This principle is based on the observation that the maximal likelihood hypothesis, given some data, is the one that minimizes the encoding length of the data as a two part code where the prefix is an optimal encoding of the hypothesis (given some prior distribution over the hypotheses), and the suffix is an optimal encoding of the data given the hypothesis. Because the length of an optimal encoding is determined by the probability distribution over the code words, data compression becomes equivalent to Bayesian induction. Usually, however, we are lacking a principled basis for assignment of prior probabilities, and so in practice the MDL principle is applied under some existing, convenient, encoding scheme. Instead of the priors determining the encoding scheme, it is the other way around: the encoding scheme determines the priors. As such, the MDL principle becomes an instantiation of Occam's razor — it reflects a bias towards simpler hypotheses.

The **no free lunch theorems** are relevant to our research because they counter any claim for a general purpose learning algorithm. Essentially, they state that only compressible (non-random) "truths" can be induced from a limited sample, and furthermore, that a learning method only does better than random sampling to the extent that it exploits some background knowledge (implicit or explicit) about the target domain [Wol 92].

2.1 Belief Nets

Belief nets (a.k.a. Bayesian networks, probability nets, causal nets) represent arbitrary joint distributions, using a network of dependencies between variables, augmented with conditional probability tables that represent the nature of those dependencies [Pea 88]. Technically, a belief net $B = \langle \mathcal{N}, \mathcal{E}, \Theta \rangle$ is a directed acyclic graph whose nodes $\mathcal{N}$ represent variables and whose directed edges $\mathcal{E}$ represent probabilistic dependencies between the variables. Associated with each node $N \in \mathcal{N}$ is a "conditional probability table" ("CPtable"), $\theta_N \in \Theta$ that provides the distribution of N's value as a function of the values assigned to N's parents. If all of the variables are binary, we can represent the CPtable for the node X, with k parents $\text{PA}(X) = \{Y_1, \ldots, Y_k\}$ by a table with 2^k rows, each row representing one possible assignment to the node's parent-set. If the i^{th} row is indexed by $\langle Y_1, \ldots, Y_k \rangle = \langle y_1, \ldots, y_k \rangle$ (where each $y_j \in \{0, 1\}$ is the value of the random parent variable Y_j), then the values for this row specify $P(\, X = 1 \,|\, \langle Y_1, \ldots, Y_k \rangle = \langle y_1, \ldots, y_k \rangle \,)$ and $P(\, X = 0 \,|\, \langle Y_1, \ldots, Y_k \rangle = \langle y_1, \ldots, y_k \rangle \,)$.[2] See Figure 1 for an example of such a structure.

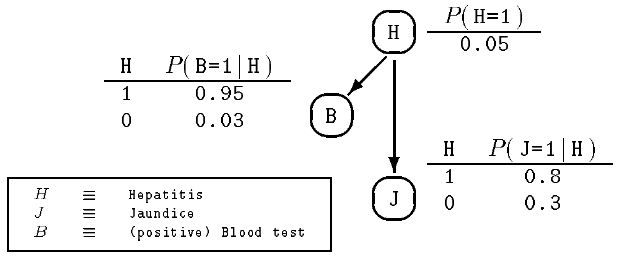

Figure 1: Simple Belief Net, B_{HBJ}

Note that B_{HBJ} represents the complete joint distribution over the three variables. While such a distribution would typically require specifying $7 = 2^3 - 1$ probabilities, this structure requires only 5. We get this saving as the structure implicitly encodes several independence claims — here, that $P(\, J \,|\, H, B \,) = P(\, J \,|\, H \,)$. (This is why B_{HBJ} does not include a link connecting `B` to `J`.) In general, the savings, realized due to the independencies encoded in the structure, can be huge; this can allow us to represent distributions over a large number of variables using only a small number of parameters [Bei 89].

Of course, a small BN can only represent a relatively small number of possible distributions (*i.e.*, only those which satisfy the independence claims of the network). A completely connected BN can represent any distribution over its variables. (This follows from the observation that any distribution can be written as $P(\, X_1, \ldots, X_n \,) = P(\, X_n \,) P(\, X_1 \,|\, X_2, \ldots, X_n \,) = \ldots = \prod_i P(\, X_i \,|\, X_{i+1}, \ldots, X_n \,)$.)

[2] Internally, we have supressed the superfluous $P(\, X = 0 \,|\, \vec{y} \,)$ values, as $P(\, X = 0 \,|\, \vec{y} \,) = 1 - P(\, X = 1 \,|\, \vec{y} \,)$.

2.2 Comparing Distributions

Most BN-learning algorithms seek a belief net that is a good match to the underlying distribution. In the standard situation, the learner does not have access to the "truth" P_T. Instead, the learner receives an empirical sample S, with associated distribution:

$$P_S(\,x\,) \quad = \quad \frac{|\{s \in S | s = x\}|}{|S|}$$

The learner seeks the the belief net that maximizes the (log) likelihood of the data given the belief net — *i.e.*, $P(\,S\,|\,B\,) \equiv P_B(\,S\,)$. As the empirical sample S is typically a degraded version of the original distribution P_T, even a learned belief net B that perfectly matches P_S will not match P_T. But, how close is B to P_T? One commonly used measure for comparing two distributions is the Kullback Leibler divergence [Kul 51]:

$$\mathrm{KLD}(\,P_T\,;\,P_B\,) \quad = \quad \sum_x P_T(\,x\,) \log\left(\frac{P_T(\,x\,)}{P_B(\,x\,)}\right) \quad (1)$$

(The "x" in the summation ranges over all $2^{O(n)}$ assignments to the variables.)[3]

To understand this measure, consider an optimal character encoding of a distribution P_T. Each of the $2^{O(n)}$ potential assignments x to the variables is an atomic event, and is assigned probability $P_T(\,x\,)$. It is well known that optimal (minimal length) encodings require $-\log P_T(\,x\,)$ bits to encode x [Cov 91]. For a given sample S, we expect to see $|S|\,P_T(\,x\,)$ occurrences of each event; therefore, we can encode (any sample S of) the entire distribution in

$$H_S \quad = \quad |S| \sum_x P_T(\,x\,) \log P_T(\,x\,)$$

bits. Now consider a distribution P_A (perhaps a belief net) which approximates P_T. If our encoder was based on these approximate probabilities, each event x would still occur with probability $P_T(\,x\,)$ but would now consume $-\log P_A(\,x\,)$ bits each. This means a sample S of the "truth" would require

$$|S| \sum_x P_T(\,x\,) \log P_A(\,x\,)$$

bits. The difference in encoding length is:

$$H_T - H_A \quad = \quad |S|\ \mathrm{KLD}(\,P_T\,;\,P_A\,)$$

bits. As the factor $|S|$ is independent of each distribution, the KL divergence (Equation 1) serves as a measure of the "accuracy" of the approximation, independent of the sample.

Note finally that, the KL divergence is always positive and achieves a minimum of zero only when $P_T(\,x\,) = P_A(\,x\,)$ for all possible events x.

[3] We assume here and throughout the paper that all logarithms are base 2.

2.3 BN-Learning Algorithms

Of the large body of results dealing with learning belief nets, the most relevant points are:

Best Structure: It is NP-hard to find the best structure in general, where "best" is based on an MDL-like measure [Chi 94]. This explains why essentially all learning algorithms search in the space of structures.

As one exception to this, there is a poly-time algorithm CL that finds the best tree-structured network [Cho 68]: Let $\mathcal{B}_{tree}$ be the set of all tree-structured belief nets (over the set of variables), where each $B \in \mathcal{B}_{tree}$ has exactly one root (with no parents), and every other node has exactly one parent. Then given any sample of complete tuples S, CL(S) will return a belief net

$$B_{CL(S)} \quad = \quad \mathrm{argmax}\{B(S) \mid B \in \mathcal{B}_{tree}\}$$

Best CPtable Entries: Given a complete set of training data, it is trivial to find the best (maximum-likelihood) CPtable entries *for a given structure*: just use the frequencies observed in the sample S [Coo 92].

For example, suppose our structure places the node A as a parent of B. Assume that the sample S includes $n_{A=0}$ tuples that assign A to 0, and $n_{B=1,A=0}$ tuples that assign B to 1 and A to 0. We would fill the CPtable entry for $P(\,B = 1\,|\,A = 0\,)$ with the ratio $n_{B=1,A=0}/n_{A=0}$.

One issue is dealing with unobserved events; *e.g.*, $n_{B=1,A=0} = 0$, or worse $n_{A=0} = 0$. In the latter case, $n_{B=1,A=0}/n_{A=0}$ is undefined. A common solution is to apply a LaPlacian correction; yielding:

$$P(\,B = 1\,|\,A = 0\,) \quad = \quad \frac{n_{B=1,A=0}\ +\ 1}{n_{A=0}\ +\ |B|} \qquad (2)$$

where $|B|$ represents the number of potential values for the attribute B.

There are straightforward extensions for finding the entries that produce the BN with the largest posterior probability, given a prior distribution over networks [Hec 95].

2.4 Minimum Description Length

Many learners attempt to learn a good "hypothesis" (*e.g.*, an accurate classification function or here, an appropriate belief net), based on a finite sample of training data. These learners typically search in a space of such hypotheses; such spaces often include hypotheses with different inherent "complexities" (*e.g.*, different number of parameters), where more complex hypotheses are able to fit the training data better than less complex ones. Unfortunately, a good fit to the training data does not guarantee good performance in general. This means a learner that simply returns the hypothesis with the best empirical fit may *overfit*.[4]

This is particularly true in our context, as a larger belief net (one with more links) will never have a worse empirical accuracy than a smaller net. Of course, given an infinite quantity of training data, we may actually want to use the *complete* net structure B_K. That is, let $\mathcal{BN}_K$ be the set of all instantations (all possible CPtables) of this B_K structure, and note this includes all possible joint distributions, and hence necessarily includes the current distribution; *i.e.*, $\mathcal{BN}_K$ includes a B^* with 0 KL divergence.

Note however that B_K (over n binary variables) requires specifying $O(2^n)$ parameters; producing good estimates for these values will require a large number of training samples. To see this, notice each training sample can only affect one CPtable entry for each node. This means we will need at least 2^{n-1} samples just to touch each entry of the CPtable of the "final" node (the one with $n-1$ parents). To produce meaningful values, of course, will require many more samples.

Applying the MDL principle is one way to address the issue of overfitting in a belief net. We can encode a data set, given a belief net, as a 2-part code, using a convenient representation for the BN and an optimal encoding of the data given the probability distribution that the belief net represents.

To describe a belief net structure, we must identify the k_N parents of each node $N \in \mathcal{N}$. Each parent can be specified using $\lceil \log |\mathcal{N}| \rceil$ bits [Cov 91]. Assuming B has binary-valued attributes, the CPtable for N contains one probability value for each of the 2^{k_N} possible assignments the parents of N. This gives us a belief net description of length

$$\mathrm{BNDL}(B) \quad = \quad \sum_{N \in \mathcal{N}} \left[\lceil \log |\mathcal{N}| \rceil k_N \; + \; d\, 2^{k_N} \right]$$

where d is a factor representing the length (in bits) of a single conditional probability value.

We can draw some observations from this result. First, more highly connected networks require longer encodings. This follows immediately from the larger conditional probability tables at each node. Another observation is that for networks with few nodes, the CPtable entries consume a very large proportion of the encoding length of the network. This follows from the fact that with few nodes, there can be only few parents as well, so $d \gg \log |\mathcal{N}|$. This suggests that $\mathrm{BNDL}(B)$ may be used as a measure of complexity for a belief net.[5]

In many cases, the probabilities are computed from a sample S, and so can be represented by $\lceil \log |S| \rceil$ bits. This estimate is improved in [Fri 96] by recognizing that the probabilities are roughly normally distributed with a variance of $\lceil \log |S| \rceil^{-\frac{1}{2}}$. Based on this normal distribution, only the low order bits are useful, and so we can encode each CP table entry computed from a sample S with

$$d \quad = \quad \frac{1}{2} \lceil \log \left(|S| \right) \rceil \tag{3}$$

bits.

The MDL principle requires us to measure the length of the data S given the belief net B. We again rely upon an optimal encoding to give us a shortest length. As noted above, we constructed the CPtables of B so that it most accurately represents the distribution. So, using B to encode each sample $s \in S$, we require $-\log(P_B(s))$ bits. Each sample element will typically occur many times in S, each distinct element x occuring with frequency:

$$f_x \quad = \quad \frac{|\{s \in S | s = x\}|}{|S|}$$

Thus, our encoding of the data will require:

$$\begin{aligned} \mathrm{DDL}(S;\, B) \quad &= \quad -\sum_s \lceil \log P_B(s) \rceil \\ &= \quad -\sum_x f_x \lceil \log P_B(x) \rceil \end{aligned}$$

bits. The second formula requires us to sum over each unique x in the sample set. Other implementations require the sum to be over the entire space of potential assignments; that computation is intractable, if the number of variables is large. Therefore, [Suz 96, Chi 97] have considered methods to eliminate the low-order marginals and reduce the cost of this summation.

The "MDL" belief net representing a sample S is the B_{MDL} that minimizes the MDL-score, which is the sum of the following two terms:

$$B_{MDL} = \operatorname*{argmin}_{B_S} \{ \mathrm{DDL}(S;\, B_S) + \mathrm{BNDL}(B_S) \} \tag{4}$$

We observe that this MDL belief net may not be unique.

As noted above, many BN-learning algorithm seek this B_{MDL}.

3 Experiments

We describe below a set of experiments designed to investigate the effectiveness of the MDL principle in

[4] Overfitting occurs when one hypothesis scores higher than another on the training data but does worse on the underlying distribution.

[5] In fact, we will use this value as the "truth complexity" of our base distribution in the experiments.

producing effective belief nets. In particular, we attempt to characterize the *true* accuracy of the belief net with the smallest (empirical) description length over a range of truth complexities and sample sizes. In addition, we varied the range of belief nets under examination, exhaustively scrutinizing all 5-node structures, and selectively sampling the space of 10-node belief net structures.

3.1 Exhaustive Study ($|\mathcal{N}| = 5$)

In these experiments, with 5 random binary variables, we exhaustively generated and evaluated all 29,281 possible BN-structures $\mathcal{BN}_5$ against 30 "true" distributions.[6] Each distribution was represented by sample sets ranging in size from 8 elements to 128 elements. Further, we average over 5 runs.

For each experiment, we randomly generated 30 belief net structures, cycling through all edge counts from 0 to $10 = \binom{5}{2}$. The random structure was completed with CPtable entries uniformly generated from $\{0/32,\ 1/32,\ \ldots,\ 32/32\}$. This gave us a "true" distribution, B_{true}, with (empirical) description length BNDL(B_{true}).[7] This BNDL(B_{true}) offers a useful measure of the "complexity" of the world we are attempting to model: it is larger for structures with more dependencies (edges).

For each sample size $k \in \{8, 16, 32, 64, 128\}$, we generated k tuples from this B_{true} distribution, $S = S(B_{true}, k)$, which we used to instantiate each structure $B \in \mathcal{BN}_5$ using frequency estimates with a LaPlacian correction (Equation 2). We then used Equation 4 to compute the MDL-score of the instantiated $B(S)$.

We then computed the (true) KL divergence KLD(B_{true}; $B(S)$) between each of these instantiated $B(S)$s and B_{true}. For comparison, we also investigated how well other "distinguished structures" fared, in particular

B_I: the independent structure (containing no links), which is necessarily the BN requiring the fewest bits to encode;

B_K: the "complete" structure (containing every possible link), which is necessarily the BN requiring the most bits to encode;

B_{CL}: the optimal tree-structured net that CL would return, containing exactly $|\mathcal{N}| - 1$ links.

For each B_{true} and value of k, we actually drew 5 samples and computed their average MDL and KLD scores. We did this for a total of 30 different "true" distributions (*i.e.*, thirty different randomly generated B_{true}s).

3.2 Stochastic Study ($|\mathcal{N}| = 10$)

It is possible that we will encounter some effect due to the fact that $|\mathcal{N}| = 5$ is relatively small. We therefore wanted to "scale" to larger classes of structures. Here, we chose $|\mathcal{N}| = 10$ variables. Unfortunately, the combinatorics prevent us from continuing to examine all possible structures. We therefore decided to stochastically sample from the space of these structures, taking care to include a wide range of "candidate" structures. That is, if we generated structures using the obvious "include each edge with probability 1/2", we would include mostly structures that had $\approx 1/2\binom{n}{2}$ of the possible arcs, and so we probably not see either the near-independent nor the near-complete structures. We therefore biased the search, to increase the chance of including at least one structure with each of $0, 1, \ldots, \binom{n}{2}$ arcs.

Here, we produced 100 different candidate graphs — call this set $\mathcal{BN}_{10}$ — making sure that this set included the special graphs mentioned above (B_I, B_K). We also computed $B_{CL}(S)$, which varies with the dataset. We then ran the same basic set of experiments. Because our $\mathcal{BN}_{10}$ is much smaller then $\mathcal{BN}_5$, we were able to perform more intensive testing. We allowed 10 data samples to be generated from each of the 100 B_{true}s.

4 Results

As stated in the previous section, our investigation involved two separate experiments, one in which we considered as hypotheses all network structures over 5 variables and one in which we sampled hypotheses from the space of 10 variable structures. In each case, we varied truth complexity and sample size, and measured true error (KL-divergence from the underlying distribution; Equation 1) as well as the MDL-score (the description length based on an error term and a network complexity term; Equation 4).

[6] While graph isomorphism is not an issue here, more than one graph structure may represent the same dependency relationships. *E.g.*, two belief nets are equivalent if they have the same arcs and the same *v*-structure. Therefore, unique belief net structures are actually acyclic graphs where compelled edges have an enforced direction, but the others can be chosen arbitrarily (but only once). Chickering [Chi 95] characterizes the equivalence classes of belief net structures, and [Chi 97] provides an $O(n^2 + e^3)$ algorithm for determining the class for a given structure. We decided not to push on this "uniqueness" issue, as it will not alter our results in any way.

[7] The (empirical) description length requires a value for d corresponding to the bit-length of a single CPtable entry. We use $d = \lceil \log 33 \rceil$, because we admit 33 different values. We could not use Equation 3 to reduce this quantity as that improvement requires the CPtable entries to be normally distributed, which is clearly false.

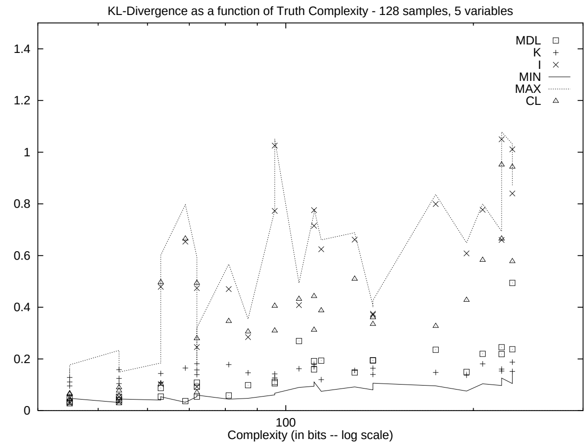

Figure 2: Results of $\mathcal{BN}_5$ with $k = 128$.

In the 5-variable networks, we see that the MDL-hypothesis outperformed the complete network across a considerable midrange of truth complexities (Figure 2)[8], given a reasonable sample size. This is the behaviour we would like to see from the MDL-hypothesis, because it suggests the heuristic is successfully handling overfitting. Note that this represents the best MDL-scoring hypothesis over *all* 5-variable networks, whereas our results for 10 variables only show the best out of 100 hypotheses generated.

On the 10-variable networks the MDL-best hypothesis did not fare as well. Given that the MDL-score depends (roughly) exponentially on the number of variables, and (roughly) linearly on the sample size, the network description length dominates the MDL-score for larger numbers of variables and smaller samples. On these 10-variable networks, the MDL-best hypothesis was often the independent network, especially when the sample size was small (see Figures 3, 4, 5).

As the sample size grew, the KL-divergences fell. For simpler truths, the KL-divergences decreased rapidly for all hypothesis networks under consideration, but the independent network performed better *for very simple truths*. As the truth got more complex, the differences between the different hypothesis types became more pronounced: the Chow-Liu and independent networks were at or near the maximum error (KL-divergence), whereas the complete network scored at or near the minimum. This was as expected: the Chow Liu and independent networks are both "dogmatic", in that they can only accomodate to evidence to a limited extent: neither converges to the truth in the limit of sample size. So we expected those networks to fare poorly when the sample size was large and the truth did not match their assumptions. However, it did not take a large sample to make this effect visible; it can be seen clearly in Figure 5, where there is only 1 datum for every two parameters of the complete network. Overall, the complete network showed a higher rate of convergence than the MDL-best network, except for very simple truths. As the sample size grew, the complete network dominated across a wider range of the complexity spectrum, and only at the extreme left (see Figures 3, 4, 5), on very simple truths, did other networks do better.

Of the networks we compared in our 10-variable experiment, (Chow Liu, MDL-best, complete, and independent) the best accuracy was attained by either the complete or independent network at most points in the space. At a certain point along the complexity spectrum they "switched places". It was interesting to see how rapid the transition was, particularly on larger samples. On the very low end of the spectrum, the independent network dominated, then it was quickly surpassed by the complete network which thereafter stayed fairly close to the minimum error, while the independent network sat on almost every error peak. There wasn't much range in between for another network type to dominate, and this gap closed as the sample size grew. The Chow Liu network was a poor performer in general; it rarely surpassed the MDL-best network in accuracy, even when that MDL-best network was the result of random sampling.

For a given sample size, as the complexity of the underlying distribution increased, the correlation between the MDL-score of a network and its KL-divergence went from positive to negative. In other words, the simpler networks were more accurate on simpler truths and less accurate on more complex truths. While this was expected, there are some additional interesting details. For simple truths, the KL-divergence of the network appears to depend logarithmically (in rough terms) on the MDL-score of the hypothesis. As the complexity of the underlying distribution grew, however, the KL-divergence became *exponentially* negatively correlated with the MDL-score. See Figures 6, 7, 8. For very small samples, the positive correlation between MDL-score and KL-divergence was present, but the negative correlation was washed out for complex truths: everything did poorly. Conversely, there was some evidence that, as sample size increased, the correlation was positive for increasingly complex truths. This effect was weak however; we did not see this on the 10-variable experiments.

MDL-scores tended to cluster at the lower end of their range (see the left side of Figures 6, 7, 8). This is not terribly surprising, since we generated the hypotheses from a flat distribution of graph sizes (in the second experiment), whereas the MDL-score is potentially ex-

[8] Section 4.1 provides a legend and summary of all of our result figures.

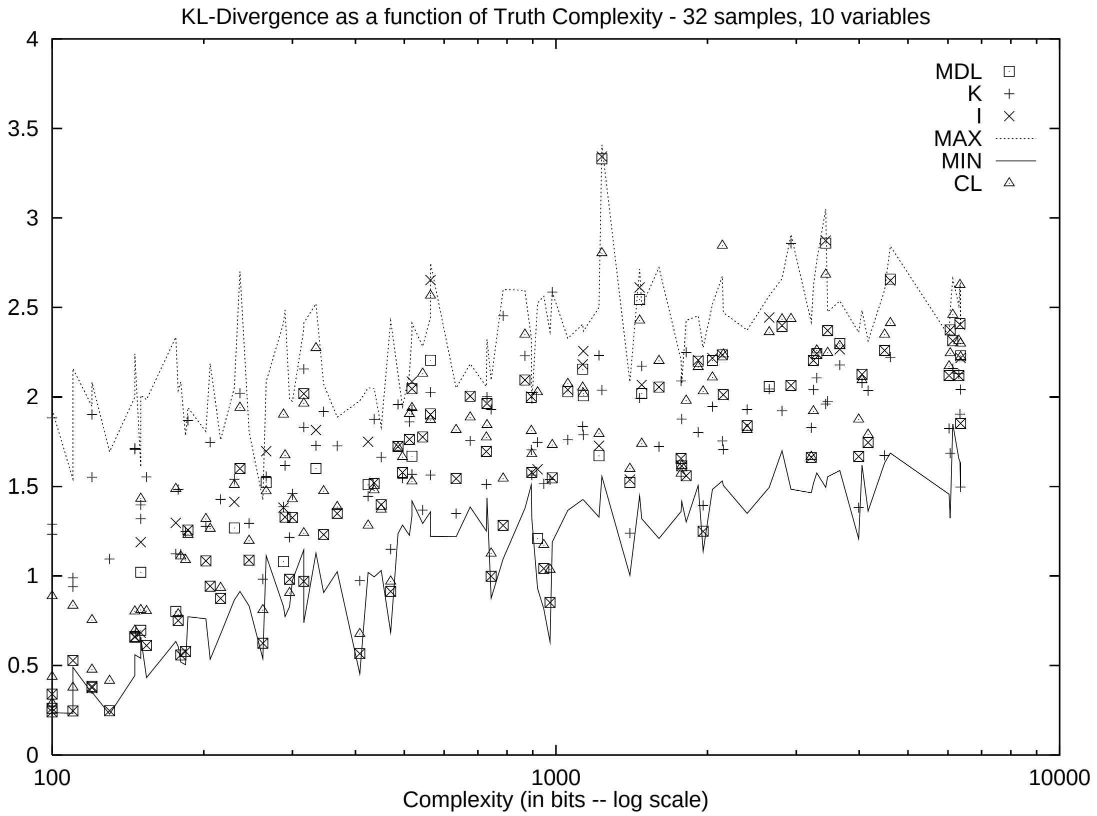

Figure 3: Results of $\mathcal{BN}_{10}$ with $k = 32$.

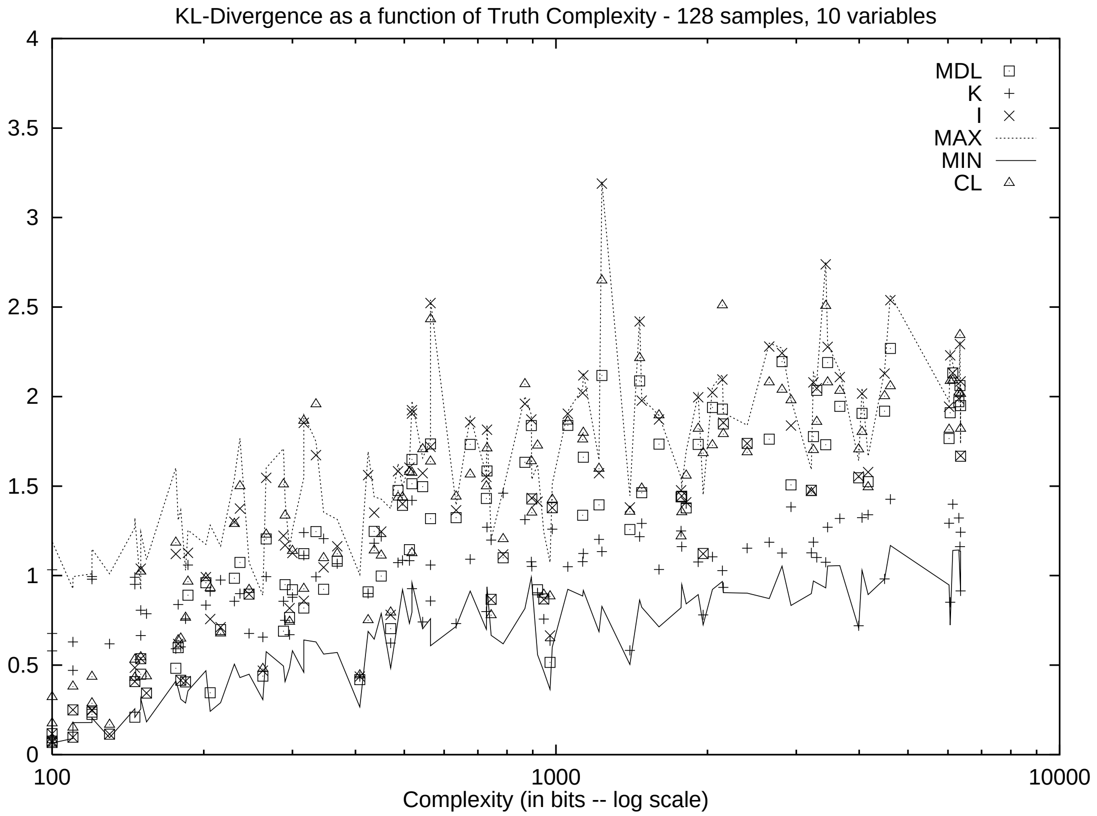

Figure 4: Results of $\mathcal{BN}_{10}$ with $k = 128$.

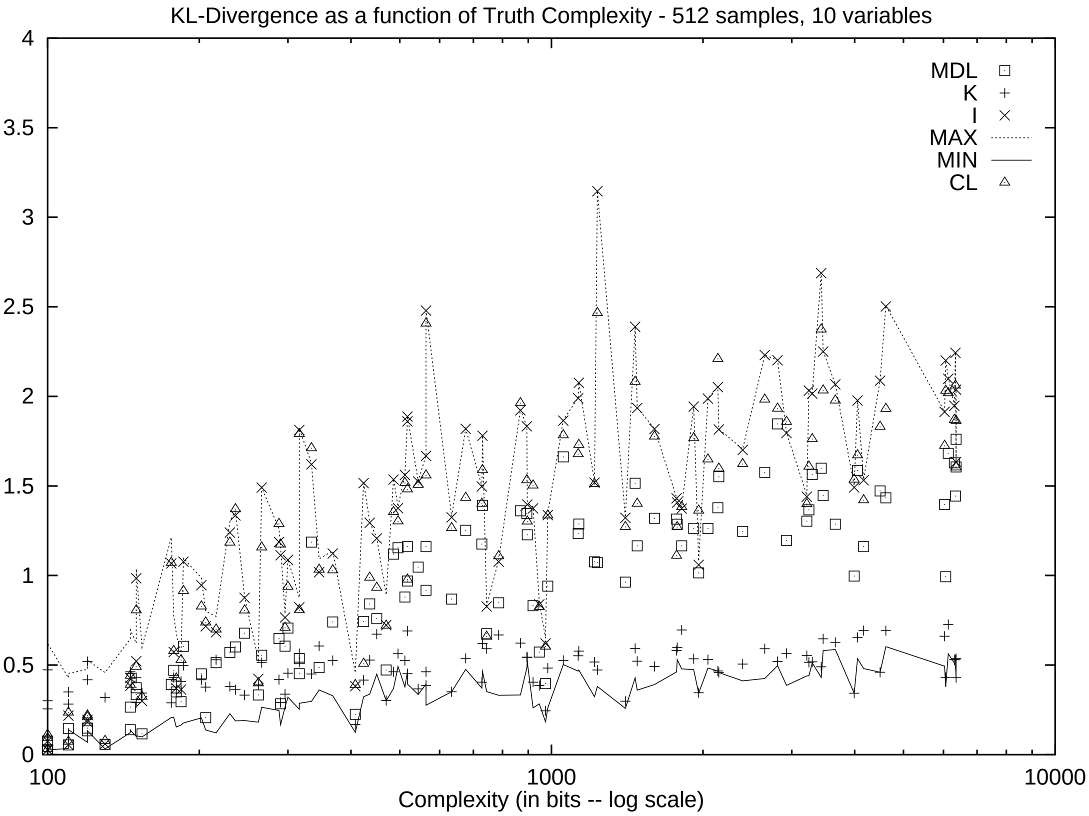

Figure 5: Results of $\mathcal{BN}_{10}$ with $k = 512$.

ponential on the graph size. But the KL-divergences were tight at this point. Taken together, this suggests that an MDL-learning algorithm might do pretty well (at finding a low MDL-score hypothesis) simply by random sampling from network structures, or alternatively, by using a branch and bound approach [Suz 96].

4.1 Description of the Figures

To conclude this summary of our results, here is a brief description of the figures we include. Note the following conventions:

(MDL) This is the hypothesis with the best MDL score, out of all hypotheses under consideration. For 5 variables, we enumerate all network structures exhaustively, so in that case MDL denotes the optimal network under the MDL score. For the 10 variable case, MDL denotes the hypothesis with the best (lowest) MDL-score out of those seen.

(K) The complete network.

(I) The independent network.

(MAX) The maximum error network.

(MIN) The minimum error network.

(CL) The Chow Liu network.

Figure 2 gives the KL-divergences of the various hypotheses under consideration, averaged over 10 randomly generated data sets, for the 5-variable case. Figures 3, 4, and 5, give a similar display for the 10 variable case, over three sample sizes. The Chow Liu network was not included in the randomly generated sample, so it may lie outside the min-max boundaries. Note that each point represents an average KL-divergence over the hypotheses that had the relevant property for a specific sample, so each may represent an average over several different network structures, except for the complete and independent networks, of course. Figures 6, 7, and 8 each show an expanded view of a single column (a single truth) from (respectively) Figures 3, 4, and 5. That is, every hypothesis is shown, with its KL-divergence plotted as a function of its MDL-score. The truth complexities for these figures are given in bits; the numbers of their parameters are 11, 90 and 273 respectively. Note, finally, the use of logarithmic scales on the x-axes.

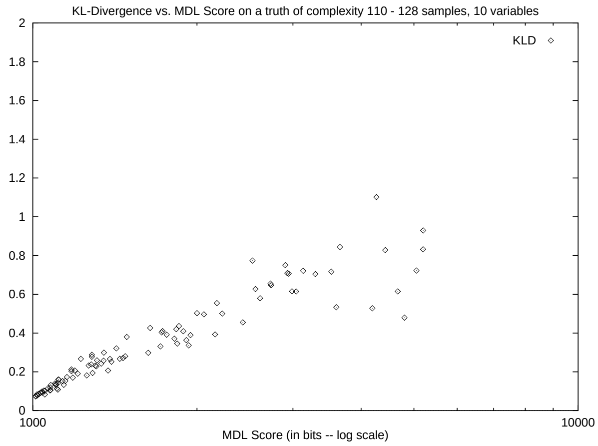

Figure 6: MDL-score vs. KLD – "simple" truth.

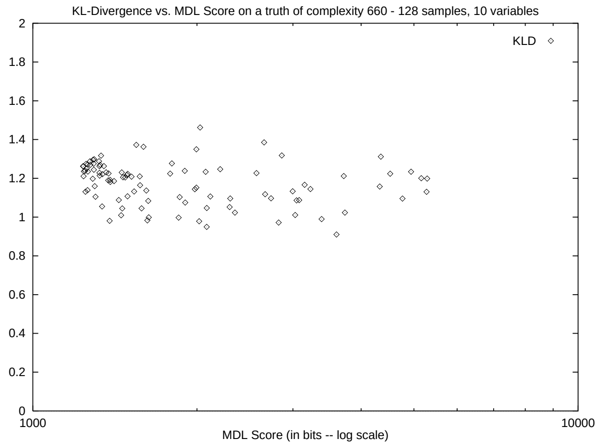

Figure 7: MDL-score vs. KLD – "intermediate" truth.

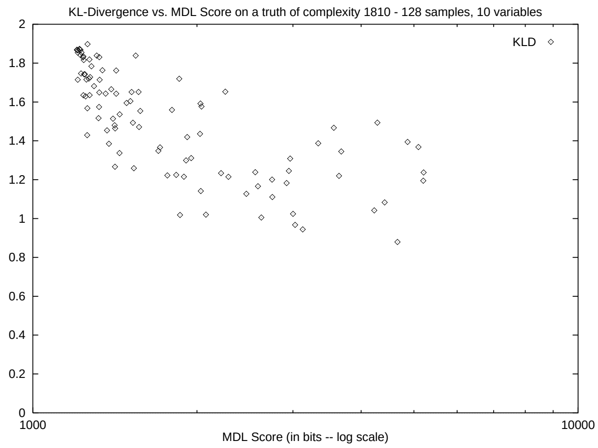

Figure 8: MDL-score vs. KLD – "complex" truth.

5 Conclusion

Our experiments give a partial characterization of where applying the MDL principle (under a specific encoding scheme) will lead to improved accuracy by avoiding overfitting. We have observed its behaviour across a range of sample sizes and truth complexities. At this point, the simplest broad characterization of MDL that we can give is that

> MDL works best when the sample size is quite large or the truth is very simple. Otherwise, it tends to *underfit* the training set.

Although we did not explore the rate of convergence of MDL (to the truth), it appears to be slower than the rate of convergence of the complete network.

One possible criticism of our investigation is its use of random network structures as a testbed. Random testbeds are considered by some to be an unfair test for heuristic methods, as they do not have the character of natural problems — *i.e.*, the distribution of problem instances in the "real world" need not corresponded to simple random generation schemes. This may be true, (in fact, it almost certainly is), but unless we can give a better characterization of what those real world distributions are like, we have little choice but to hunt for them in a larger space. Our goal in this research is not to claim superiority or inferiority for any algorithm or cost function, but to understand better the assumptions behind those heuristics and algorithms, and make them explicit.

The results reported in this paper are preliminary – further analysis of the data is needed to make our claims more precise. We plan to explore further the relationship between MDL and accuracy, on larger spaces of hypotheses, bigger data sets, and under other encoding schemes. We are also interesting in theoretically deriving the relationships between sample size, truth complexity, hypothesis complexity and accuracy.

The strength and weakness of the MDL approach is the need to specify the priors in terms of an encoding scheme for the hypotheses. It is a learning framework that must be instantiated with an encoding scheme to be meaningful. The choice of an appropriate encoding scheme is thus a critical determinant of the success or failure of an MDL approach to learning. Our results seem to indicate that, under the encoding scheme we used, the MDL-score is too strongly biased toward simple networks. We suggest weighting the error term more heavily, to make MDL more accurate (at the expense of accepting larger networks). Also, we suggest different encoding schemes for belief nets be considered. The measure of network complexity used here was somewhat naive, in that it considers only the dependency structure of the network, but not the nature of those dependencies. CPtables might be represented by some more concise representation, such as decision trees where the leaves are distributions [Fri 96a]. Last, we propose detaching MDL from its theoretical framework, given that that framework rests on an unrealistic assumption (an optimal encoding for hypotheses) to allow for modifications that make it more appropriate for a given learning task.

References

[Bei 89] I.A. Beinlich, H.J. Suermondt, R.M. Chavez, and G.F. Cooper, The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks, *Proceedings of the 2nd European Conference on AI*, Springer Verlag, 1989.

[Blu 86] A. Blumer, A. Ehrenfeucht D. Haussler, and M.K. Warmuth, Occam's Razor, *Information Processing Letters*, Vol. 24, 1997.

[Chi 94] D.M. Chickering, D. Geiger, and D.E. Heckerman, Learning Bayesian Networks is NP-Hard, *Microsoft Research Technical Report MSR-TR-94-17*, 1994.

[Chi 95] D.M. Chickering, A Transformational Characterization of Equivalent Bayesian Network Structures, *UAI'95*, 1995.

[Chi 97] D.M. Chickering and D.E. Heckerman, Efficient Approximations for the Marginal Likelihood of Bayesian Networks with Hidden Variables, *Machine Learning*, Vol. 29, 1997.

[Cho 68] C.K. Chow and C.N. Liu, Approximating Discrete Probability Distributions with Dependence Trees, *IEEE Transactions on Information Theory*, Vol. 14, 1968.

[Coo 92] G. Cooper and E. Herskovits, A Bayesian Method for the Induction of Probabilistic Networks from Data", *Machine Learning*, Vol. 9, 1992.

[Fri 96a] N. Friedman and M. Goldszmidt, Learning Bayesian Networks with Local Structure, *UAI'96*, 1996.

[Fri 96] N. Friedman and Z. Yakhini, On the Sample Complexity of Learning Bayesian Networks, *UAI'96*, 1996.

[Hec 95] D.E. Heckerman, A Tutorial on Learning With Bayesian Networks, *Microsoft Research Technical Report MSR-TR-95-06*, revised 1996.

[How 97] C. Howson, A Logic of Induction, *Philosophy of Science*, Vol. 64, 1997.

[Kul 51] S. Kullback and R.A. Leibler, On information and sufficiency, *Annals of Mathematics and Statistics*, Vol. 22, 1951.

[Lam 94] W. Lam and F. Bacchus, Learning Bayesian Belief Networks: An approach base on the MDL Principle, *Computational Intelligence*, Vol. 10, 1994.

[Cov 91] T.M. Cover and J.A. Thomas, *Elements of Information Theory*, Wiley, 1991.
McGraw-Hill, 1997.

[Mur 95] P.M. Murphy and M.J. Pazzani, Exploring the Decision Forest: An Empirical Investigation of Occam's Razor in Decision Tree Induction, *JAIR*, 1994.

[Pea 88] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann, 1988.

[Ris 85] J. Rissanen, Minimum Description Length Principle, *Encyclopædia of Statistical Sciences 5*, Wiley, 1985.

[Suz 96] J. Suzuki, Learning Bayesian Belief Networks Based On The Minimum Description Length Principle: An Efficient Algorithm Using the Branch and Bound Technique", *Machine Learning*, 1996.

[Val 84] L.G. Valiant, A Theory of the Learnable, *Communications of the ACM*, Vol. 27, 1984.

[Wol 92] D.H. Wolpert, On Overfitting Avoidance as Bias, *Santa Fe Institute Technical Report SFI-TR-92-03-5001*, 1992.