

# An approximated MDL score for learning Bayesian networks

Josep Roure Alcobé <sup>1</sup>

Escola Universitària Politècnica de Matarn  
Av. Puig i Cadafalch 101-111  
08303 Matarn, Spain

## Abstract

In this paper we present an approximation to the mutual information among a single variable and a set of variables. The main aim of our approach is to reduce the amount of sufficient statistics (i.e. frequency counters) required to calculate the mutual information. To do so, we use the chain rule and assume different independency statements among variables. We will use our approximation to calculate the MDL of a given Bayesian network. We will show that our approximated approach to the MDL measure is score equivalent and we will use it in order to learn Bayesian networks from data. We will experimentally see that learning algorithms that use our approach obtain high quality Bayesian networks. We also note that our approach can be used in any information based measures.

## 1 Introduction

We will present, in this work, an approximation to the mutual information  $I(\mathbf{X}; Y)$  among a single variable  $Y$  and a set of variables  $\mathbf{X}$  that avoids estimating, from data, joint probabilities of large set of variables. We will use our approach to calculate the MDL score and learn Bayesian networks.

It is widely reported in the *statistical pattern recognition* literature, see (Jain et al., 2000), that the performance of a classifier depends on the interrelationship between sample sizes, number of features, and classifier complexity. It has been often observed in practice that adding variables to a classifier may actually degrade its performance if the number of data instances that are used to learn the classifier is small relative to the number of variables. This is known as the *peaking phenomenon* which is a consequence of the *curse of dimensionality* (Jain et al., 2000), usually stated as follows: in order to estimate a joint probability, the number of required data instances grows exponentially with the number of variables. This is due to the fact that the required number of parameters in order to estimate a joint probability distribution

grows exponentially with the number of variables, i.e the number of counters of a contingency table. This is also illustrated by (Hastie et al., 2001), when they state that the sampling density is proportional to  $N^{1/p}$ , where  $p$  is the number of variables and  $N$  is the sample size. Thus, if  $N_1 = 100$  represents a dense sample for a single input problem, then  $N_{10} = 100^{10}$  is the sample size required for the same sampling density with 10 inputs. Thus in high dimensions all feasible training samples sparsely populate the input space.

The main aim of our approach is to reduce the memory required to store *sufficient statistics*. This is very important in some learning environments. In *incremental* environments, where new data are processed as long as they are available without re-processing the previously learned ones, it is required to store all the *sufficient statistics* (Roure, 2004). In this sort of environments, in order to scale up learning algorithms, it is very important to reduce the amount *sufficient statistics* required.

Also algorithms that learn from very large data streams, see (Hulten and Domingos, 2002) can benefit from our approach. In this environment, learning algorithms attempt to minimize

the number of data instances used to produce the model. For doing so, algorithms iteratively learn models increasing the number of data instances used and stop when they observe that the model quality does not grow when additional data is used. Some works, like (Meek et al., 2002), observe that to gather the number of data to be used it succeeds to use an approximated algorithm that is cheap in computing time, and to learn the final model with the *full* learning algorithm using the appropriate number of data instances.

In this paper, we will use the chain rule to obtain another expression for the mutual information that has two desirable properties. Firstly, the expression is incremental in the number of variables, that is, we will be able to express the mutual information when a new variable  $Z$  is added to the set of variables  $\mathbf{X}$  as the mutual information among  $Y$  and  $\mathbf{X}$  plus the information due to the variable  $Z$ . Secondly, the obtained expression will allow us to drastically reduce the number of variables considered at each term but still take into account all the relationships among pairs of variables in  $\mathbf{X}$  and variable  $Y$ . The first property is useful for Bayesian network learning since algorithms build networks from arc-less structures by incrementally adding variables to the sets of parents.

The rest of the paper is organized as follows. In the rest of this section we introduce Bayesian networks and a well-known learning algorithm. We also introduce the MDL based quality measure for Bayesian networks that uses the mutual information among a variable  $X_i$  and the set of its parents  $\mathbf{Pa}_i$ . In Section 2 we will use the chain rule to obtain the above mentioned expression for the mutual information. In Section 3 we will introduce the approximated mutual information and compare it to the exact one. In Section 4 we will use our approach to obtain an approximated MDL measure and we will show that it is *score equivalent*. Finally, we give some experimental results.

### 1.1 Learning Bayesian networks

A *Bayesian network* is an annotated directed acyclic graph that encodes a joint probabil-

ity distribution of a set of random variables  $\mathbf{X} = \{X_1, \dots, X_n\}$  each of which has a domain of possible values. Formally, a Bayesian network for  $\mathbf{X}$  is a pair  $BN = (B_S; B_P)$  where the first component,  $B_S$ , is a directed acyclic graph (DAG) whose vertices correspond to the random variables  $X_1, \dots, X_n$ , and whose edges represent directed dependencies between variables. The parents of  $X_i$ , denoted as  $\mathbf{Pa}_i$ , is the set of variables with an arc to  $X_i$  in the graph. The model structure yields to a factorization of the joint probability distribution for  $\mathbf{X}$ ,  $P(\mathbf{X}) = \prod_{i=1}^n P(X_i | \mathbf{Pa}_i)$ . The second component,  $B_P$ , represents the parameters that quantifies the network. It has a parameter  $\mu_{ijk} = P(X_i = x_i^k | \mathbf{Pa}_i = \mathbf{pa}_i^j)$  for each possible state  $x_i^k$  of  $X_i$  and for each configuration  $\mathbf{pa}_i^j$  of  $\mathbf{Pa}_i$ .

Most of the learning algorithms found in the literature are hill-climbing searchers that begin with the arc-less network and perform the operator that most increases the score of the resulting structure and does not introduce a cycle into the network. Algorithms stop when the use of a single operator cannot increase the network's score. The difference between the algorithms is the domain of models and the operators they use.

For our experiments we will use algorithm B (Buntine, 1991) that yields full DAG structures and the neighborhood of a given network structure,  $B_S$ , is the set of all networks that can be obtained from  $B_S$  by adding a single arc  $X_i \rightarrow X_j$  to  $B_S$  such that does not introduce a cycle,

$$\mathcal{N}_H(B_S) = \{(\mathbf{X}, E') | E' = E \cup \{(X_i, X_j)\} \wedge B'_S \text{ is a DAG}\}$$

In order to measure the quality of the alternative structures we will use the MDL quality measure that we explain in the following subsection.

### 1.2 MDL Scoring function

MDL approach to scoring functions for Bayesian networks is based on the idea that the best model of a database is the model that minimizes the sum of the encoding length of the model plus the encoding length of the data given

the model. (Friedman and Goldszmidt, 1996) used  $-\log P_B(\mathbf{u})$  as an approximation of the encoding length of each instance  $\mathbf{u}$ , and obtained the following expression for the encoding length,  $DL(D|B)$ , of the whole dataset  $D$  given the Bayesian network  $B$ ,

$$DL(D|B) = - \sum_{i=1}^N \log P_B(\mathbf{u}) \quad (1)$$

where  $N$  is the number of data instances in  $D$ . They transformed Equation (1) and obtained the following equivalent one,

$$\begin{aligned} DL(D|B) &= N \sum_{i=1}^n H(X_i | \mathbf{Pa}_i) \\ &= N \sum_{i=1}^n H(X_i) - I(X_i; \mathbf{Pa}_i) \end{aligned}$$

where  $n$  is the number of variables,  $H(X)$  is the entropy of variable  $X$  and  $I(X; Y)$  is the mutual information among variables  $X$  and  $Y$ . Another approach (Lam and Bacchus, 1994) took the Kullback-Leibler divergence as a measure of the encoding length of the data given the Bayesian network arriving to a similar expression.

The encoding length of the network structure is usually expressed as the total number of parameters that we need to store for the network. Note that for each variable  $X_i$  we need to store  $|\mathbf{Pa}_i|(|X_i| - 1)$  parameters. The number of bits used for each of these parameters is usually taken to be  $1=2 \log N$ , and so, the encoding length,  $K(B)$ , for the whole Bayesian network structure is

$$K(B) = \frac{1}{2} \log N \sum_{i=1}^n |\mathbf{Pa}_i|(|X_i| - 1)$$

Note that the MDL measure is factored, or equivalently, the sum property holds. That is, the encoding length of the whole Bayesian network is expressed as the sum of the encoding length of each variable and its parent set.

$$MDL(B|D) = \sum_{i=1}^n MDL(X_i; \mathbf{Pa}_i)$$

This property is very important for learning algorithms since it localizes the effect of an addition (or removal) of an arc to the families affected (i.e. a variable and its parent set).

This property also holds for the Bayesian approach to quality measures (Cooper and Herskovits, 1992).

Another property that is usually required for scoring measures is that they give the same quality score to Bayesian network structures that are equivalent, that is, structures that define the same probability distribution. When this property holds for a given quality measure it is said to be *score equivalent*. Note that the MDL approach to quality functions is *score equivalent* (Chickering, 1995).

## 2 Incremental Mutual Information

The mutual information  $I(\mathbf{X}; Y)$  and the joint entropy  $H(\mathbf{X}; Y)$  can be incrementally expressed using the chain rule (Cover and Thomas, 1991). Let  $\mathbf{X}^{(n)} = \{X_1; X_2; \dots; X_n\}$

$$I(\mathbf{X}^{(n)}; Y) = I(X_1; Y) + I(X_2; Y|X_1) + \dots + I(X_n; Y|\mathbf{X}^{(n-1)})$$

which can be expressed as

$$I(\mathbf{X}^{(n)}; Y) = I(X_1; Y) + \sum_{i=2}^n I(X_i; Y|\mathbf{X}^{(i-1)}) \quad (2)$$

Note that this expression is incremental in the number of variables, that is, when a new variable,  $Z$ , is added to a set of variables,  $\mathbf{X}$ , we can express the mutual information as

$$I(\mathbf{X}^{(n)} Z; Y) = I(\mathbf{X}^{(n)}; Y) + I(Z; Y|\mathbf{X}^{(n)})$$

Now we will further decompose the mutual information using the following identity  $I(X; Y|Z) = I(X; Y) + I(X; Z|Y) - I(X; Z)$  and reordering terms, from Equation (2) we obtain

$$\begin{aligned} I(\mathbf{X}^{(n)}; Y) &= \sum_{i=1}^n I(X_i; Y) + \\ &\sum_{i=2}^{n-1} I(\mathbf{X}^{(i-1)}; X_i|Y) + I(\mathbf{X}^{(n-1)}; X_n|Y) \quad (3) \\ &\sum_{i=2}^{n-1} -I(\mathbf{X}^{(i-1)}; X_i) - I(\mathbf{X}^{(n-1)}; X_n) \quad (4) \end{aligned}$$

Using the chain rule and the identity stated above to the last terms of Equation (3) and (4) we obtain

$$\begin{aligned}
I(\mathbf{X}^{(n)}; Y) &= \sum_{i=1}^n I(X_i; Y) + \sum_{i=1}^{n-1} [I(X_i; X_n | Y) - I(X_i; X_n)] + \\
&\quad \sum_{i=2}^{n-2} I(\mathbf{X}^{(i-1)}; X_i | Y X_n) + I(\mathbf{X}^{(n-2)}; X_{n-1} | Y X_n) \\
&\quad - \sum_{i=2}^{n-2} -I(\mathbf{X}^{(i-1)}; X_i) - I(\mathbf{X}^{(n-2)}; X_{n-1})
\end{aligned}$$

Performing repeatedly the same substitutions, we obtain

$$\begin{aligned}
I(\mathbf{X}^{(n)}; Y) &= \sum_{i=1}^n I(X_i; Y) \\
&\quad + \sum_{i=2}^n \sum_{j=1}^{i-1} I(X_i; X_j | Y X_n \dots X_{i+1}) \\
&\quad - \sum_{i=2}^n \sum_{j=1}^{i-1} I(X_i; X_j | X_n \dots X_{i+1})
\end{aligned}$$

and where the incremental expression is

$$\begin{aligned}
I(\mathbf{X}^{(n)} Z; Y) &= I(\mathbf{X}^{(n)}; Y) \\
&\quad + I(Z; Y) \tag{5}
\end{aligned}$$

$$\begin{aligned}
&\quad + \sum_{i=1}^n I(Z; X_i | Y X_n \dots X_{i+1}) \tag{6}
\end{aligned}$$

$$\begin{aligned}
&\quad - \sum_{i=1}^n I(Z; X_i | X_n \dots X_{i+1}) \tag{7}
\end{aligned}$$

Let us take a look to this last expression. We can see that the contribution to the mutual information among a set of variables  $\mathbf{X}^{(n)}$  and a variable  $Y$  when a new variable,  $Z$ , is added to  $\mathbf{X}^{(n)}$  can be divided into three parts. The first one, term (5), measures the contribution of the new variable to the mutual information with  $Y$ . The second one, term (6), measures the relationship between  $Z$ ,  $\mathbf{X}^{(n)}$  and  $Y$ . This term grows when the mutual information among  $Z$  and  $\mathbf{X}^{(n)}$  given  $Y$  grows. The third one, term (7), decreases when the mutual information among  $Z$  and  $\mathbf{X}^{(n)}$  increases.

To take a closer look let us consider the terms (6) and (7) when  $i = n$ ,  $I(X_n; Z | Y) - I(X_n; Z)$ . We note that this term may be positive or negative (McKay, 1999): when the joint probability distribution of variables forms a **Markov chain**

(see Figure 2 (a)), then  $I(X_n; Z | Y) \leq I(X_n; Z)$ . Further note that, in this case,  $I(X_n; Z | Y) = 0$  and that  $I(Y; X_n) \geq I(Z; X_n)$ . On the contrary, when the joint probability distribution forms a **V-structure** (see Figure 2 (b)), then  $I(X_n; Z | Y) \geq I(X_n; Z)$ , and also that  $I(X_n; Z | Y)$  is maximum. A typical example of this joint probability structure is *The Burglar Alarm* problem (Pearl, 1988).

So, from the considerations stated above, we can see that the contribution of a new variable  $Z$  to the mutual information  $I(\mathbf{X}^{(n)} Z; Y)$ :

- The higher the mutual information among  $Y$  and the new variable  $Z$  is, the higher its contribution to the whole mutual information,  $I(\mathbf{X}^{(n)} Z; Y)$ , is.
- The more similar the joint probability distribution of  $\mathbf{X}^{(n)}$ ,  $Z$ , and  $Y$  is to a V-structure the higher the contribution of  $Z$  to the whole mutual information is.
- The higher the mutual information among  $Z$  and  $\mathbf{X}^{(n)}$  is, the lower its contribution to the whole mutual information is.

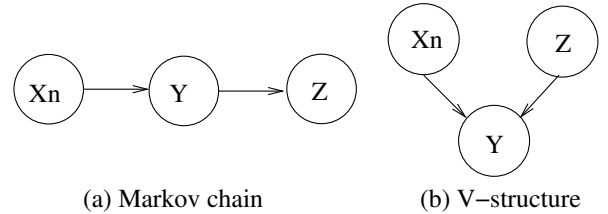


Figure 1: Joint probability distribution structures

We also want to stress that  $I(\mathbf{X}^{(n)}; Y) \leq I(\mathbf{X}^{(n)} Z; Y)$  (Cover and Thomas, 1991). This can easily be seen in Equation (2) where each variable contributes with a non-negative amount since the mutual information is always non-negative. For notational simplicity we will note as  $\mathbb{C}(Z)$  the sum of terms (5), (6) and (7).

### 3 Approximated mutual information

In this section we present an approximation to the mutual information. The aim of our approximation is to drastically reduce the number of

variables involved in the joint probabilities in order to reduce the memory space used to store the *sufficient statistics*.

To avoid using joint probabilities of large set of variables in the mutual information  $I(\mathbf{X}^{(n)}; Y)$ , we will discard conditioning the mutual information with variables in  $\mathbf{X}^{(n)}$ ,

$$I(\mathbf{X}^{(n)}; Z; Y) = I(\mathbf{X}^{(n)}; Y) + I(Z; Y) \quad (8)$$

$$+ \sum_{i=1}^n I(Z; X_i | Y) \quad (9)$$

$$- \sum_{i=1}^n I(Z; X_i) \quad (10)$$

Note that in our approximation we are calculating joint probabilities of set of three variables at most. This reduce the amount of *sufficient statistics* (e.g. counters) used from  $k^{n+2}$ , in the *full* version of the mutual information, to  $k^3$ , where  $k$  is the number of values a variable can take and  $n$  is the number of variables involved:  $\mathbf{X}^{(n)}; Z; Y$ .

In our approximation, we are assuming in each term  $i$  that the variable  $X_i$  is independent of the set of variables  $X_n; \dots; X_{i+1}$ . Obviously terms (9) and (10) will, in general, give different results than the corresponding terms, (6) and (7), of the *full* mutual information. Observe though, that this new expression keeps the three contributions of a new variable  $Z$  to the mutual information that measures different aspects of the dependencies among the variables, or groups of variables, involved.

We will note as  $\Phi_a(Z)$  the sum of terms (8), (9) and (10). See that while  $0 \leq \Phi(Z)$ , it does not hold for  $\Phi_a(Z)$ . Furthermore, observe that the approximated measure may be either equal, greater or lower than the exact measure. For example, when the underlying joint probability of  $Z$ ,  $\mathbf{X}^{(n)}$  and  $Y$  forms a Markov chain such as  $Z \rightarrow \mathbf{X}^{(n)} \rightarrow Y$ , then  $\Phi(Z) = 0$  and  $\Phi_a(Z) = I(Z; Y) + \sum_{i=1}^n [I(Z; Y | X_i) - I(Z; Y)]$  and since in this case  $I(Z; Y | X_i) \ll I(Z; Y)$ ,  $\Phi_a(Z) \leq 0$ .

Unfortunately, we have not been able to give an analytical bound for the error introduced by our approximated mutual information yet. However, we have experimentally seen

that when variables form a Markov chain then  $\Phi_a(Z) \leq \Phi(Z)$ , and when variables form a V-structure being  $Y$  at the vertex then  $\Phi_a(Z) \geq \Phi(Z)$  most of the times. Observe that the more independent the variables in  $\mathbf{X}^{(n)}$  are, the more similar  $\Phi(Z)$  and  $\Phi_a(Z)$  will be.

## 4 Approximated MDL

In this section we will use our approximated mutual information in order to obtain an approximated MDL measure. First we express the mutual information in an incremental way. Given a variable,  $X_i$ , and its parent set,  $\mathbf{Pa}_i = \{Pa_{i1}; \dots; Pa_{in}\}$ , when a new parent,  $Z$ , is added the MDL can be expressed as

$$MDL(X_i, \mathbf{Pa}_i; Z) = K(X_i, \mathbf{Pa}_i) + DL(X_i, \mathbf{Pa}_i) \quad (11)$$

$$+ (\frac{1}{2} \log N) |\mathbf{Pa}_i| (|X_i| - 1) (|Z| - 1) \quad (12)$$

$$- NI(Z; X_i) \quad (13)$$

$$- N \sum_{j=1}^n I(Z; Pa_{ij} | X_i Pa_{in} \dots Pa_{ij+1}) \quad (14)$$

$$+ N \sum_{j=1}^n I(Z; Pa_{ij} | Pa_n \dots Pa_{ij+1}) \quad (15)$$

Note that in this equation we state the *MDL* for a given variable  $X_i$  and its parent set  $\{\mathbf{Pa}_i\} \cup \{Z\}$ . The *MDL* for the whole Bayesian network structure can be easily obtained summing the *MDL* over all variables and their parent sets, since the *MDL* is factored. See that the term (11) corresponds to  $MDL(X_i; \mathbf{Pa}_i)$ , term (12) corresponds to the new encoding length when a parent is added and the last three terms, (13), (14) and (15), correspond to  $-N\Phi(Z)$ .

Now we state our approximated version of the *MDL* using the notation from the former section,

$$MDL_a(X_i, \mathbf{Pa}_i; Z) = K(X_i, \mathbf{Pa}_i) + DL_a(X_i, \mathbf{Pa}_i) \quad (16)$$

$$+ (\frac{1}{2} \log N) |\mathbf{Pa}_i| (|X_i| - 1) (|Z| - 1) \quad (17)$$

$$- NI(Z; X_i) \quad (18)$$

$$- N \sum_{j=1}^n I(Z; Pa_{ij} | X_i) \quad (19)$$

$$+N \sum_{j=1}^n I(Z; Pa_{ij}) \quad (20)$$

where  $ML_a(X_i; \mathbf{Pa}_i)$  corresponds to the encoding length of data given the network structure using our approximation to mutual information. Note again that the last three term of the equation { (18), (19) and (20) } correspond to  $-N\Phi_a(Z)$ .

Observe that since a *good* parent set of a variable is the one that minimizes the MDL score, our approximated measure will, in general, over score the contribution of a variable  $Z$  when it is a *good* parent of  $X_i$  given  $\mathbf{Pa}_i$  and will under score its contribution when it is a *bad* parent. This is due to the fact the approximated mutual information is lower than the exact one when  $Z$  forms a Markov chain with  $\mathbf{Pa}_i$  and  $X_i$ , and it is higher when  $Z$  forms a V-structure.

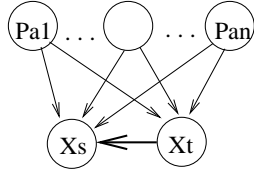


Figure 2:  $X_s \leftarrow X_t$  is a covered arc

Now we will show that our approach is *score equivalent*. First, we need to define the notion of *covered arc*: An arc,  $Y \rightarrow X$  in a network structure is *covered* if  $\mathbf{Pa}_x = \mathbf{Pa}_y \cup \{Y\}$ , see Figure 4. It is seen that, between any pair of equivalent Bayesian networks, there exists a sequence of distinct covered arc reversals that make both structures identical (Chickering, 1995). Thus, to show that a scoring function is *score equivalent* it suffices to see that it gives the same score to two structures that differ in a single covered arc reversal.

**Lemma 1** *Let  $U$  be a set of variables, and let  $D$  be a database over  $U$ . Let  $B_1$  and  $B_2$  be two network structures over  $U$ . Furthermore, let  $X_s$  and  $X_t$  be two nodes in  $B_1$  and  $B_2$ , where  $\mathbf{Pa}_{X_s} = \{Pa_1; \dots; Pa_n\}$  and  $\mathbf{Pa}_{X_t} = \{Pa_1; \dots; Pa_n\} \cup \{X_s\}$  in  $B_1$  and  $\mathbf{Pa}_{X_s} = \{Pa_1; \dots; Pa_n\} \cup \{X_t\}$  and  $\mathbf{Pa}_{X_t} = \{Pa_1; \dots; Pa_n\}$  in  $B_2$ , and the parent sets for the rest of variables in  $U$  are the same in both structures. Then,*

$$MDL_a(B_1, D) = MDL_a(B_2, D)$$

**Proof:** Since our approach measures in the same way the number of parameters of Bayesian networks than the *full* approach, we only need to show that the scoring length of the data given both structures,  $B_1$  and  $B_2$ , is the same,  $DL(B_1; D) = DL(B_2; D)$ .

$$\begin{aligned} DL_a(B_1, D) &= DL_a(B_2, D) \Leftrightarrow \\ &ML_a(X_s, \{Pa_1, \dots, Pa_n\}) + \\ &ML_a(X_t, \{Pa_1, \dots, Pa_n\}) + \Delta_a(X_s) \\ &= \\ &ML_a(X_s, \{Pa_1, \dots, Pa_n\}) + \Delta_a(X_t) + \\ &ML_a(X_t, \{Pa_1, \dots, Pa_n\}) \\ &\Leftrightarrow \\ &\Delta_a(X_s) = \Delta_a(X_t) \\ &\Leftrightarrow \\ &I(X_t; X_s) + \sum_{i=1}^n I(X_s; Pa_i | X_t) - I(X_s; Pa_i) \\ &= \\ &I(X_s; X_t) + \sum_{i=1}^n I(X_t; Pa_i | X_s) - I(X_t; Pa_i) \\ &\Leftrightarrow \\ &\forall i \in [1, \dots, n] I(X_s; Pa_i | X_t) - I(X_s; Pa_i) = \\ &I(X_t; Pa_i | X_s) - I(X_t; Pa_i) \\ &\Leftrightarrow \\ &\sum_{X_s X_t Pa_i} P(X_s X_t Pa_i) \log \frac{P(X_s X_t Pa_i) P(Pa_i) P(X_s) P(X_t)}{P(Pa_i X_t) P(X_s X_t) P(Pa_i X_s)} \\ &= \\ &\sum_{X_s X_t Pa_i} P(X_s X_t Pa_i) \log \frac{P(X_s X_t Pa_i) P(Pa_i) P(X_s) P(X_t)}{P(Pa_i X_t) P(X_s X_t) P(Pa_i X_s)} \\ &\square \end{aligned}$$

## 5 Experimental results

In this section we compare the performance of algorithm B (see Section 1.1) when it uses the *full MDL* and the approximated *MDL<sub>a</sub>*. We used the datasets Adult (48.842 instances and 13 variables), Car (1.728 inst. and 7 var.), DNA (3.190 inst. and 61 var.), Letter Recognition (20.000 inst. and 16 var.) Mushroom (8.124 inst. and 23 var.) and Nursery (12.960 inst. and 9 var.) from the UCI machine learning repository (Murphy and Aha, 1994), the Alarm dataset (20.000 inst. and 37 var.) (Cooper and Herskovits, 1992) that is a standard benchmark in the Bayesian network literature and a synthetic dataset that were kindly donated by Robert Castelo (Castelo and Koeka, 2003),

that we will call Synthetic (100.000 inst. and 25 var.). We used the discretization utility of MLC++ (Kohavi et al., 1994) in order to dis-

data given the Bayesian network. Experiments confirmed this last point since the Bayesian networks obtained with the approximated MDL, in general, had more arcs.

We think that there is still a lot of work to be done in order to further understand the error introduced with our approximation. We will quantitatively study the error introduced by our approach and to study in which situations, underlying probability distributions of data, this error is highest and lowest. We would also like to use our approximation to other non hill climbing strategies to learn Bayesian networks (Tian, 2000) and to relate our approach to others like (Friedman and Getoor, 1999).

### Acknowledgments

This work has been partially supported by FEDER and CICYT, TIC-2003-08382-C05-02.

### References

- W. Buntine. 1991. Theory refinement on Bayesian networks. In B.D. D'Ambrosio, P. Smets, and P.P. Bonisone, editors, *Proceedings of the 7th UAI*.
- Robert Castelo and Tomás Koeka. 2003. On inclusion-driven learning of Bayesian networks. *Journal of Machine Learning Research*, (4), September.
- D. M. Chickering. 1995. A transformational characterization of equivalent Bayesian networks. In P. Besnard and S. Hanks, editors, *Conference on Uncertainty in Artificial Intelligence*, pages 87{98. Morgan-Kaufman.
- G. Cooper and E. Herskovits. 1992. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309{347.
- T. M. Cover and J. A. Thomas. 1991. *Elements of Information Theory*. Wiley series in telecommunications. John Wiley & Sons, Inc.
- N. Friedman and L. Getoor. 1999. Efficient learning using constrained sufficient statistics. In *International Workshop on Artificial Intelligence*.
- N. Friedman and M. Goldszmidt. 1996. Learning Bayesian networks with local structure. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence*.
- T. Hastie, R. Tibshirani, and J. Friedman. 2001. *The elements of statistical learning. Datamining, inference and prediction*. Springer Series in Statistics. Springer.
- G. Hulten and P. Domingos. 2002. Mining complex models from arbitrarily large databases in constant time. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 525{531. ACM Press.
- Anil K. Jain, Robert P. W. Duin, and Jianchang Mao. 2000. Statistical pattern recognition: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1):4{37.
- R. Kohavi, G. John, R. Long, D. Manley, and K. Pöeger. 1994. MLC++: A Machine Learning library in c++. In *Proceedings of the Sixth International Conference on Tools with Artificial Intelligence*, pages 740{743. IEEE Computer Society Press.
- W. Lam and F. Bacchus. 1994. Learning Bayesian belief networks. an approach based on the MDL principle. *Computational Intelligence*, 10(4):269{293.
- D. J.C. McKay. 1999. *Information theory, inference and learning algorithms*. <http://wol.ra.phy.cam.ac.uk/mackay/itprnn/book.ps.gz>.
- Christopher Meek, Bo Thieson, and David Heckerman. 2002. The learning-curve sampling method applied to model-based clustering. *Journal of Machine Learning Research*, (2).
- P.M. Murphy and D.W. Aha. 1994. UCI repository of Machine Learning databases. <http://www.ics.uci.edu/mllearn/MLRepository.html>. Irvine, CA: University of California, Department of Information and Computer Science.
- J. Pearl. 1988. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann.
- J. Roure. 2004. Incremental augmented naive bayes classifiers. In *Proceedings of the sixteenth European Conference of Artificial Intelligence (ECAI 2004)*. IOS Press.
- Jin Tian. 2000. A branch-and-bound algorithm for MDL learning bayesian networks. In *Proceedings of Uncertainty in Artificial Intelligence*, pages 580{588. Morgan Kaufmann.