Bayesian Network Structure Learning using Factorized NML Universal Models

Teemu Roos, Tomi Silander, Petri Kontkanen, and Petri Myllymäki
Complex Systems Computation Group, Helsinki Institute for Information Technology HIIT
University of Helsinki & Helsinki University of Technology
P.O.Box 68 (Department of Computer Science)
FIN-00014 University of Helsinki, Finland
Email: firstname.lastname@cs.helsinki.fi

***Abstract*— Universal codes/models can be used for data compression and model selection by the minimum description length (MDL) principle. For many interesting model classes, such as Bayesian networks, the minimax regret optimal normalized maximum likelihood (NML) universal model is computationally very demanding. We suggest a computationally feasible alternative to NML for Bayesian networks, the factorized NML universal model, where the normalization is done locally for each variable. This can be seen as an approximate sum-product algorithm. We show that this new universal model performs extremely well in model selection, compared to the existing state-of-the-art, even for small sample sizes.**

I. Introduction

The stochastic complexity of a sequence under a given model class is a central concept in the minimum description length (MDL) principle [1], [2], [3], [4]. Its interpretation as the length of the shortest achievable encoding makes it a yardstick for the comparison of different model classes. In recent formulations of MDL, stochastic complexity is defined using the so called normalized maximum likelihood (NML) universal model, originally introduced by Shtarkov [5] for data compression; for the role of NML in MDL model selection, see [6], [7], [3], [4], [8].

Since the introduction of the NML universal model in the context of MDL, there has been significant interest in the evaluation of NML stochastic complexity for different practically relevant model classes, both exactly and asymptotically. For discrete models, exact evaluation is often computationally infeasible since it involves a normalizing coefficient which is a sum over all possible data-sets. For continuous cases, the normalizing coefficient is an integral which can be solved in only a few cases. Under certain conditions on the model class, different versions of stochastic complexity (which include two-part, mixture, and NML forms) have the same asymptotic form, the so called Fisher information approximation. However, for small data-sets and for model classes that do not satisfy the necessary conditions, the asymptotic form is not accurate [9].

Exact and computationally tractable formulas are rare: results for multinomial models are given in [10], and for Bayesian networks with structural restrictions in [11], [12], [13]; more references can be found in [3] and [4].

In this paper, we introduce the *factorized NML* (fNML) universal model for Bayesian networks. The rest of the paper is organized as follows: In Sections II and III we discuss the normalized maximum likelihood (NML) and sequentially normalized maximum likelihood (sNML) models, respectively. In Section IV we review the basics of Bayesian networks. The factorized NML model is introduced in Section V, where it is also shown to be computationally feasible for all Bayesian networks. Finally, in Section VI, we present experimental results, demonstrating that fNML compares favorably in a model selection task, relative to the current state-of-the-art.

II. Normalized Maximum Likelihood Models

Before describing the sequential NML and factorized NML models, we fix some notation and review some basic properties of the well-known NML model. Let

$$x^n := \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,m} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,m} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_{1,:} \\ \mathbf{x}_{2,:} \\ \vdots \\ \mathbf{x}_{n,:} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_{:,1} \mathbf{x}_{:,2} \cdots \mathbf{x}_{:,m} \end{pmatrix} ,$$

be an $n \times m$ data matrix where each row, $\mathbf{x}_{i,:} = (x_{i,1}, x_{i,2}, \ldots, x_{i,m})$, $1 \leq i \leq n$, is an m-dimensional observation vector, and columns of x^n are denoted by $\mathbf{x}_{:,j}$, $1 \leq j \leq m$.

A parametric probabilistic model $\mathcal{M} := \{p(x^n \; ; \; \theta) \; : \; \theta \in \Theta\}$, where Θ is a parameter space, assigns a probability mass or density value to the data. A *universal model* for $\mathcal{M}$ is a single distribution that, roughly speaking, assign almost as high a probability to any data as the the maximum likelihood parameters $\hat{\theta}(x^n)$.

Formally, model $\hat{p}(x^n)$ is 'universal' (in the point-wise sense) if and only if it satisfies

$$\lim_{n \to \infty} \frac{1}{n} \ln \frac{p(x^n \; ; \; \hat{\theta}(x^n))}{\hat{p}(x^n)} = 0 \; , \tag{1}$$

i.e., the log-likelihood ratio, often called the 'regret', is allowed to grow sublinearly in the sample size n. The celebrated *normalized maximum likelihood* (NML) universal model [5],

[6] is given by

$$p_{\mathrm{NML}}(x^n) := \frac{p(x^n \; ; \; \hat{\theta}(x^n))}{C_n}$$
$$C_n = \int_{\mathcal{X}^n} p(x^n \; ; \; \hat{\theta}(x^n)) \, dx^n \;\; .$$

It is the unique minimax optimal universal model in the sense that the worst-case regret is minimal. In fact, it directly follows from the definition that the regret is a constant dependent only on the sample size n:

$$\ln \frac{p(x^n \; ; \; \hat{\theta}(x^n))}{p_{\mathrm{NML}}(x^n)} = \ln C_n \;\; .$$

For some model classes, the normalizing factor is finite only if the range $\mathcal{X}^n$ of the data is restricted, see e.g. [6], [14], [15]. For discrete models, the normalizing constant, C_n, is given by a sum over all data matrices of size $n \times m$:

$$C_n = \sum_{x^n \in \mathcal{X}^n} p(x^n \; ; \; \hat{\theta}(x^n)) \;\; .$$

The practical problem arising in applications of the NML universal model is then to evaluate the normalizing constant. For continuous models the integral can be solved in closed form for only a few specific models. For discrete models, the time complexity of the naive solution, i.e., summing over all possible data matrices, grows exponentially in both n and m, and quickly becomes intractable. Even the second-most naive solution, summing over equivalence classes of matrices, sharing the same likelihood value, is usually intractable even though often polynomial in n.

The usual Fisher information approximation [6]

$$\ln C_n = \frac{k}{2} \ln \frac{n}{2\pi} + \ln \int_{\Theta} \sqrt{\det I(\theta)} \, d\theta + o(1) \;\; ,$$

where k is the dimension of the parameter space, is also non-trivial to apply due to the integral involving the Fisher information $I(\theta)$. Using only the leading term (with or without 2π), i.e., the BIC criterion [16], gives a rough approximation which, as a rule, performs worse in model selection tasks than more refined approximations or, ideally, the exact solution, see e.g. [3, Chap. 9].

III. Sequentially Normalized ML Models

A recent family of variants of NML, called the *sequentially* (or *conditional*) *normalized maximum likelihood* (sNML) [17], [4] has similar minimax properties like NML but is often significantly easier to use in practice.

For data matrix $x^n = (\mathbf{x}_{1,:}, \mathbf{x}_{2,:}, \ldots, \mathbf{x}_{n,:})'$, the sNML-1 model is defined as

$$p_{\mathrm{sNML1}}(x^n) := \prod_{i=1}^{n} \frac{p(\mathbf{x}_{i,:} \mid x^{i-1} \; ; \; \hat{\theta}(x^i))}{K_i(x^{i-1})} \;\; , \tag{2}$$

$$K_i(x^{i-1}) := \int p(\mathbf{x}_{i,:} \mid x^{i-1} \; ; \; \hat{\theta}(x^i)) \, d\mathbf{x}_{i,:} \;\; , \tag{3}$$

where normalization ensures that each factor in the product is a proper density function.

There is also another variant of sNML, which we call here sNML-2. It can be defined in analogy with (2) as follows:

$$p_{\mathrm{sNML2}}(x^n) := \prod_{i=1}^{n} \frac{p(x^i \; ; \; \hat{\theta}(x^i))}{K'_i(x^{i-1})} \;\; , \tag{4}$$
$$K'_i(x^{i-1}) := \int p(x^i \; ; \; \hat{\theta}(x^i)) \, d\mathbf{x}_{i,:} \;\; .$$

Using the sNML-2 model is equivalent to predicting the ith observation using the standard NML model defined for sequences of length i. Formally we have

$$p_{\mathrm{NML}}(\mathbf{x}_{i,:} \mid x^{i-1}) = p_{\mathrm{sNML2}}(\mathbf{x}_{i,:} \mid x^{i-1}) \;\; .$$

Note that the standard NML model is not in general a stochastic process, which makes it possible that

$$p_{\mathrm{NML}}(\mathbf{x}_{i,:} \mid x^{i-1}) \neq \sum_{\mathbf{x}_{i+1,:}} p_{\mathrm{NML}}(\mathbf{x}_{i,:}, \mathbf{x}_{i+1,:} \mid x^{i-1}) \;\; , \tag{5}$$

and hence, typically two NML models, defined for sequences of different lengths, give different predictions. In contrast, both sNML-1 and sNML-2 are by definition stochastic processes, so that for them we always have an equality in (5).

Regrets Visualized. Figure 1 gives a visualization of the regrets of four universal models in the Bernoulli case: the Laplace predictor ("add-one"), the Krichevsky–Trofimov predictor ("add-half"), sNML-2, and NML. For NML, the initial sequence probabilities, $q(x^t)$, are obtained from a fixed NML model, defined for $n = 5$, by summing over the possible continuations of length $n-t$. For the Bernoulli model, sNML-1 is equivalent to the Laplace predictor.

Related Work. The sNML-2 model has been analysed earlier in conjuction with discrete Markov models, including as a special case the Bernoulli model, by Shtarkov [5] (see his Eq. 45). Also, Takimoto and Warmuth [18] analyze a slightly more restricted minimax problem, the solution of which agrees with sNML-2 for Markov models. Grünwald [4] uses the term "conditional NML" (CNML) for a family of universal models, conditioned on an initial sequence without considering the joint model obtained as a product of such conditional densities. Our sNML-1 corresponds to his CNML-3, and our sNML-2 corresponds to his CNML-2. The conditional mixture codes studied by Liang and Barron [19] are also closely related to sNML, and have similar minimax properties.

IV. Bayesian Networks

We will next, in Sec. V, describe a new NML variant, similar to the sNML models discussed in the previous section. This new variant gives a computationally feasible universal model, and a corresponding model selection criterion, for general Bayesian network models. This section presents the necessary background in Bayesian networks.

First, let us associate with the columns, $\mathbf{x}_{:,1}, \ldots, \mathbf{x}_{:,m}$, a directed acyclic graph (DAG), $\mathcal{G}$, so that each column is represented by a node. Each node, $X_j, 1 \leq j \leq m$, has a (possibly empty) set of *parents*, Pa_j, defined as the set of nodes with an outgoing edge to node X_j. Without loss of generality, we require that all the edges are directed towards

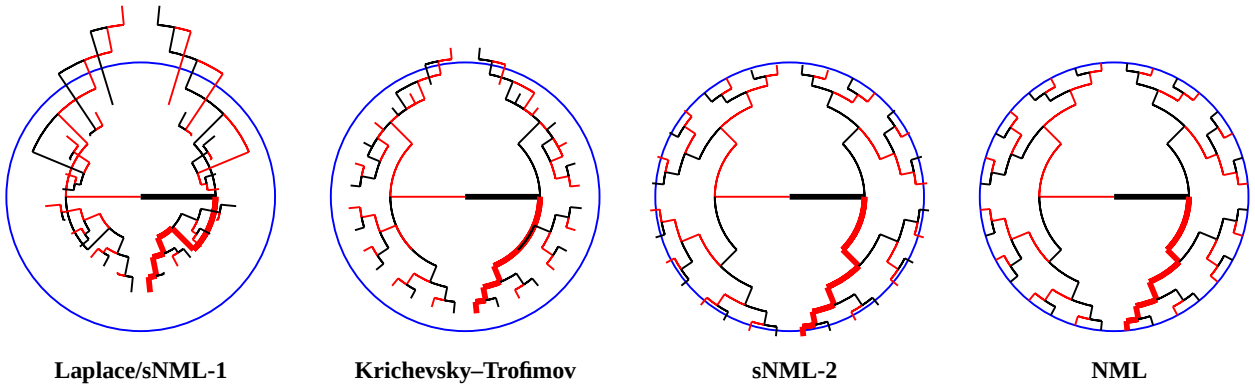

Fig. 1. Regrets of four universal models in the Bernoulli case. Each path from the origin (center) to the boundary represents a binary sequence of length $n = 5$. Red edges correspond to `1`s, black edges to `0`s. The path for sequence `01111` is emphasized. The distances from the origin of the branching points are given by the regrets $\ln[p(x^t \,;\, \hat{\theta}(x^t))/q(x^t)]$ for each prefix x^t. The blue circle shows the regret of NML. For the Bernoulli model, Laplace and sNML-1 coincide. Note the similarity between sNML-2 and NML.

increasing node index, i.e., $\mathrm{Pa}_j \subseteq \{1, \ldots, j-1\}$. Figure 2 gives an example.

The idea is to model dependencies among the nodes (i.e. columns) by defining the joint probability distribution over the nodes in terms of *local distributions*: each local distribution specifies the conditional distribution of each node given its parents, $p(X_j \mid \mathrm{Pa}_j), 1 \leq j \leq m$. It is important to notice that these are *not* dependencies among the subsequent rows of the data matrix x^n, but dependencies 'inside' each row, $\mathbf{x}_{i,:}, 1 \leq i \leq n$. Indeed, in all of the following, we assume that the rows are independent realizations of a fixed (memoryless) source.

The local distributions can be modeled in various ways, but here we focus on the discrete case. The probability of a child node taking value $x_{i,j} = r$ given the parent nodes' configuration, $\mathrm{pa}_{i,j} = \mathbf{s}$, is determined by the parameter

$$\theta_{j|\mathrm{Pa}_j}(r, \mathbf{s}) = p(x_{i,j} = r \mid \mathrm{pa}_{i,j} = \mathbf{s} \,;\, \theta_{j|\mathrm{Pa}_j}) \ ,$$

for all $1 \leq i \leq n, 1 \leq j \leq m$, where the notation $\theta_{j|\mathrm{Pa}_j}(r, \mathbf{s})$ refers to the component of the parameter vector $\theta_{j|\mathrm{Pa}_j}$ indexed by the value r and the configuration $\mathbf{s}$ of the parents of X_j. For empty parent sets, we let $\mathrm{pa}_{i,j} \equiv 0$. For instance, consider the graph of Fig. 2; on each row, $1 \leq i \leq n$, the parent configuration of column $j = 8$ is the vector $\mathrm{pa}_{i,8} = (x_{i,1}, x_{i,5}, x_{i,7})$; the parent configuration of column $j = 1$ is $\mathrm{pa}_{i,1} = 0$, etc.

The joint distribution is obtained as a product of local distributions:

$$p(x^n \,;\, \theta) = \prod_{j=1}^{m} p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \,;\, \theta_{j|\mathrm{Pa}_j}) \ . \tag{6}$$

This type of probabilistic graphical models are called Bayesian networks [20]. Factorization (6) entails a set of conditional independencies, characterized by so called Markov properties, see [21]. For instance, the *local Markov property* asserts that each node is independent of its non-descendants given its parents, generalizing the familiar Markov property of Markov chains.

It is now possible to define the NML model based on (6) and a fixed graph structure $\mathcal{G}$:

$$p_{\mathrm{NML}}(x^n \,;\, \mathcal{G}) = \frac{\prod_{j=1}^{m} p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \,;\, \hat{\theta}(x^n))}{C_n} \ , \tag{7}$$

$$C_n = \sum_{x^n} \prod_{j=1}^{m} p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \,;\, \hat{\theta}(x^n)) \ . \tag{8}$$

The required maximum likelihood parameters are easily evaluated since it is well known that the ML parameters are equal to the relative frequencies:

$$\hat{\theta}_{j|\mathrm{Pa}_j}(r, \mathbf{s}) = \frac{\big|\{i \,:\, x_{i,j} = r, \mathrm{pa}_{i,j} = \mathbf{s}\}\big|}{\big|\{i' \,:\, \mathrm{pa}_{i',j} = \mathbf{s}\}\big|} \ , \tag{9}$$

where $|S|$ denotes the cardinality of set S. However, as pointed out in Sec. II, summing over all possible data matrices is not tractable except in toy problems where n and m are both very small. Efficient algorithms have been discovered only recently for restricted graph structures [11], [12], [13].

V. Factorized NML Models

As a computationally less demanding alternative to NML in the context of Bayesian networks, we define the *factorized NML* (fNML) in a similar spirit as sNML. We let the joint probability distribution be given by a product of *locally* normalized maximum likelihood distributions:

$$p_{\mathrm{fNML}}(x^n \,;\, \mathcal{G}) := \prod_{j=1}^{m} \frac{p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \,;\, \hat{\theta}(x^n))}{Z_j(\mathrm{Pa}_j)} \tag{10}$$

$$= \frac{\prod_{j=1}^{m} p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \,;\, \hat{\theta}(x^n))}{Z(x^n)} \ , \tag{11}$$

where

$$Z_j(\mathrm{Pa}_j) = \sum_{X'_j} p(X'_j \mid \mathrm{Pa}_j \,;\, \hat{\theta}(X'_j, \mathrm{Pa}_j)) \tag{12}$$

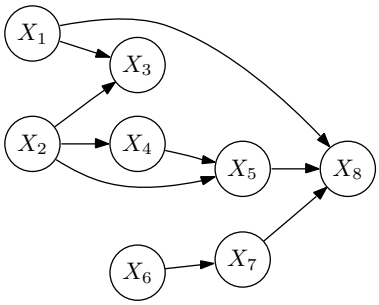

Fig. 2. An example of a directed acyclic graph (DAG). The parents of node X_8 are $\{X_1, X_5, X_7\}$. The descendants of X_4 are $\{X_5, X_8\}$.

is a sum over all possible instantiations of column $\mathbf{x}_{:,j}$, and

$$Z(x^n) = \prod_{j=1}^{m} \sum_{X'_j} p(X'_j \mid \mathrm{Pa}_j \; ; \; \hat{\theta}(X'_j, \mathrm{Pa}_j)) \qquad (13)$$

is the product of the local normalizing factors. The local normalizing factors $Z_j(\mathrm{Pa}_j)$ can be decomposed further into simple multinomial NML normalization constants, one for each parent configuration in Pa_j. Using the recently discovered linear-time algorithm [10] for the multinomial case, the total computation time becomes feasible even for large sample sizes and for many variables (columns).

Note that, as can be seen from (9), the maximum likelihood parameters of each local distribution, $\theta_{j|\mathrm{Pa}_j}$, depend only on column $\mathbf{x}_{:,j}$ and column(s) Pa_j. In particular, since we require $\mathrm{Pa}_j \subseteq \{1, \ldots, j-1\}$, we have

$$\begin{aligned} &p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \; ; \; \hat{\theta}(x^n)) = p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \; ; \; \hat{\theta}(\mathbf{x}_{:,1}, \ldots, \mathbf{x}_{:,j})) \\ &= p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \; ; \; \hat{\theta}(\mathbf{x}_{:,j}, \mathrm{Pa}_j)) \;\; , \end{aligned} \qquad (14)$$

of which the second form, where only the first j columns appear, is the one that should be used in (10) by analogy with (2). Due to the above identity, the expressions can be used interchangeably.

The sum-product view. It is interesting to compare the NML and fNML models. Consider Eqs. (7) and (11): the constant normalizer of NML, C_n, an exponential *sum of products*, is replaced in fNML by $Z(x^n)$, a *product of sums* that depends on the data. The fNML model can therefore be seen as 'cheating' by using a sum-product algorithm, where the distributive law (see [22])

$$\begin{aligned} \begin{cases} f(x_1, x_2) \equiv f(x_1) \\ g(x_1, x_2) \equiv g(x_2) \end{cases} \quad &\Longrightarrow \quad \sum_{x_1, x_2} f(x_1, x_2) g(x_1, x_2) \\ &= \left(\sum_{x_1} f(x_1) \right) \left(\sum_{x_2} g(x_1) \right) \end{aligned} \qquad (15)$$

is applied to compute the sum in C_n even though the terms do not actually factor column-wise into independent parts. No cheating is necessary when the graph is empty, i.e., when $\mathrm{Pa}_j = \emptyset$ for all $1 \leq j \leq m$. This means that we have $Z(x^n) = C_n$, which by (7) and (11) implies that for empty graphs p_{NML} and p_{fNML} are equivalent.

The regrets of the two models are easily seen to be $\ln C_n$ and $\ln Z(x^n)$, for NML and fNML respectively. Notice also that the regret of fNML, $\ln Z(x^n)$, depends on the data only through the parents, $\mathrm{Pa}_j, 1 \leq j \leq m$, and hence, is independent of all the leaf nodes, i.e., nodes that have no descendants. Again, if the graph is empty, all nodes are leafs and $Z(x^n) = C_n$ for all x^n so that the NML and fNML models are equivalent.

Finally, we observe that for fNML the two variants of sNML, sNML-1 and sNML-2, coincide. Letting $x(j) := (\mathbf{x}_{:,1}, \mathbf{x}_{:,2}, \ldots, \mathbf{x}_{:,j})$ denote the first j columns, we obtain

$$\begin{aligned} p(x(j) \; ; \; \hat{\theta}(x(j))) &= \prod_{l=1}^{j} p(\mathbf{x}_{:,l} \mid \mathrm{Pa}_l \; ; \; \hat{\theta}(\mathbf{x}_{:,l}, \mathrm{Pa}_l)) \\ &= p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \; ; \; \hat{\theta}(x^n)) \prod_{l=1}^{j-1} p(\mathbf{x}_{:,l} \mid \mathrm{Pa}_l \; ; \; \hat{\theta}(\mathbf{x}_{:,l}, \mathrm{Pa}_l)) \;\; , \end{aligned}$$

where both equalities depend on (14). The last factor on the right-hand side is independent of column $\mathbf{x}_{:,j}$. When the above is normalized with respect to $\mathbf{x}_{:,j}$, this factor cancels and we are left with $p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \; ; \; \hat{\theta}(x^n))$, which exactly what is normalized in (10). Hence, it doesn't matter whether we define fNML as in (10) or as the product over $1 \leq j \leq m$ of the normalized versions of $p(x(j) \; ; \; \hat{\theta}(x(j)))$.

VI. Experiment

To empirically test performance of the fNML-criterion in Bayesian network structure learning task, we generated several Bayesian networks, and then studied how different model selection criteria succeeded in learning the model structure from data. The most often used selection criterion for the task is the Bayesian Dirichlet Equivalence score [23], but due to its sensitivity to the choice of prior hyperparameter, we chose two different versions of it: $BDe_{0.5}$ and $BDe_{1.0}$. We also included the Bayesian Information Criterion, BIC. All these scores can be interpreted to implement some version of the MDL-criterion.

In the following, we present the results for an experiment in which we generated 1800 different Bayesian network models, which we tried to learn back using the data generated from these models. We generated the networks using 5, 10 and 15 variables, and also varied the number of arcs and the parameters of the networks. We then generated 1000, 10000 and 10000 data vectors from each network, and tried to learn the models back using these data samples and different scoring criteria. It turned out that learning the models back with these sample sizes was practically possible only for smallest networks containing 5 nodes. Varying the number of arcs and the parameters did not seem to have a strong effect on the outcome. This made it possible to concentrate on comparing the performance of different scoring criteria for different sample sizes (Figure 3).

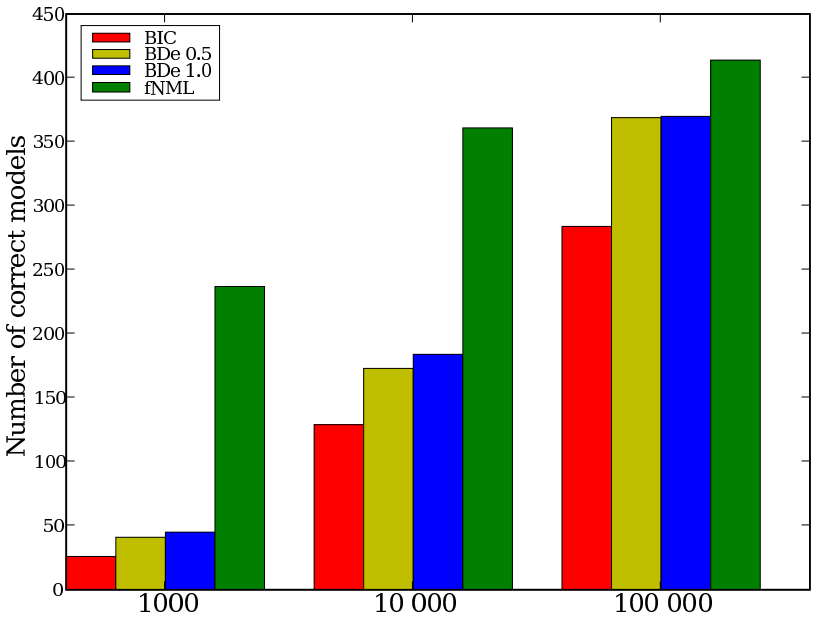

Fig. 3. Number of correctly learned models in 1800 trials for different sizes of data and different scoring criteria.

The results clearly show that fNML excels with small sample sizes. With large sample sizes, the difference is not that big, which is hardly surprising, since asymptotically, they all converge to the data generating model. This result is significant, since BDe score(s) can be regarded as the current state-of-the-art. Furthermore, the fNML score is computationally no more demanding than the BDe score.

ACKNOWLEDGMENT

This work was supported in part by the Finnish Funding Agency for Technology and Innovation under projects KUKOT and PMMA, by the Academy of Finland under project CIVI, and by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778.

REFERENCES

[1] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 445–471, 1978.

[2] ——, "Stochastic complexity," *Journal of the Royal Statistical Society*, vol. 49, no. 3, pp. 223–239 and 252–265, 1987.

[3] ——, *Information and Complexity in Statistical Modeling*. Springer, 2007.

[4] P. Grünwald, *The Minimum Description Length Principle*. MIT Press, 2007.

[5] Y. Shtarkov, "Universal sequential coding of single messages," *Problems of Information Transmission*, vol. 23, pp. 3–17, 1987.

[6] J. Rissanen, "Fisher information and stochastic complexity," *IEEE Transactions on Information Theory*, vol. 42, no. 1, pp. 40–47, January 1996.

[7] ——, "Strong optimality of the normalized ML models as universal codes and information in data," *IEEE Transactions on Information Theory*, vol. 47, no. 5, pp. 1712–1717, July 2001.

[8] J. Myung, D. Navarro, and M. Pitt, "Model selection by normalized maximum likelihood," *Journal of Mathematical Psychology*, vol. 50, pp. 167–179, 2006.

[9] D. Navarro, "A note on the applied use of MDL approximations," *Neural Computation*, vol. 16, no. 9, pp. 1763–1768, 2004.

[10] P. Kontkanen and P. Myllymäki, "A linear-time algorithm for computing the multinomial stochastic complexity," *Information Processing Letters*, vol. 103, no. 6, pp. 227–233, 2007.

[11] P. Kontkanen, H. Wettig, and P. Myllymäki, "NML computation algorithms for tree-structured multinomial Bayesian networks," *EURASIP Journal on Bioinformatics and Systems Biology*, 2008 (in press).

[12] H. Wettig, P. Kontkanen, and P. Myllymäki, "Calculating the normalized maximum likelihood distribution for Bayesian forests," in *Proc. IADIS International Conference on Intelligent Systems and Agents*, Lisbon, Portugal, July 2007.

[13] T. Mononen and P. Myllymäki, "Fast NML computation for naive Bayes models," in *Proc. 10th International Conference on Discovery Science*, Sendai, Japan, October 2007.

[14] J. Rissanen, "MDL denoising," *IEEE Transactions on Information Theory*, vol. 46, no. 7, pp. 2537–2543, 2000.

[15] S. de Rooij and P. Grünwald, "An empirical study of minimum description length model selection with infinite parametric complexity," *Journal of Mathematical Psychology*, vol. 50, no. 2, pp. 180–192, 2006.

[16] G. Schwarz, "Estimating the dimension of a model," *Annals of Statistics*, vol. 6, pp. 461–464, 1978.

[17] J. Rissanen and T. Roos, "Conditional NML models," in *Proc. 2007 Information Theory and Applications Workshop (ITA-07)*, San Diego, CA, January–February 2007, pp. 337–341.

[18] E. Takimoto and M. Warmuth, "The last-step minimax algorithm," in *Proc. 11th International Conference on Algorithmic Learning Theory*, 2000, pp. 279–290.

[19] F. Liang and A. Barron, "Exact minimax strategies for predictive density estimation, data compression, and model selection," *IEEE Transactions on Information Theory*, vol. 50, no. 11, pp. 2708–2726, 2004.

[20] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, San Mateo, CA, 1988.

[21] S. Lauritzen, *Graphical Models*. Oxford University Press, 1996.

[22] S. M. Aji and R. J. McEliece, "The generalized distributive law," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 325–343, 2000.

[23] D. Heckerman, D. Geiger, and D. Chickering, "Learning Bayesian networks: The combination of knowledge and statistical data," *Machine Learning*, vol. 20, no. 3, pp. 197–243, September 1995.