Proceedings of the

Recent Breakthroughs in Minimum Description Length Learning Workshop

An ICML/UAI/COLT Workshop

Helsinki, 9 July, 2008

Organizing Committee

Tim van Erven	CWI, Amsterdam	Tim.van.Erven@cwi.nl
Peter Grnwald	CWI, Amsterdam	pdg@cwi.nl
Petri Myllymki	University of Helsinki	Petri.Myllymaki@cs.helsinki.fi
Teemu Roos	Helsinki Institute for Information Technology	Teemu.Roos@cs.helsinki.fi
Ioan Tabus	Tampere University of Technology	tabus@tut.fi

Acknowledgements

The organizing committee would like to thank the ICML/COLT/UAI workshop organizers: Sanjoy Dasgupta, Nando de Freitas, John Langford and Michael Littman.

This workshop was supported in part by the IST Programme of the European Community, under the PASCAL-2 Network of Excellence. This publication only reflects the authors' views.

Preface

During the last few years (2004-2007), there have been several breakthroughs in the area of Minimum Description Length (MDL) modeling, learning and prediction. These breakthroughs concern the efficient computation and proper formulation of MDL in parametric problems based on the "normalized maximum likelihood", as well as altogether new, and better, coding schemes for nonparametric problems. This essentially solves the so-called AIC-BIC dilemma, which has been a central problem in statistical model selection for more than 20 years now. The goal of this workshop is to introduce these exciting new developments to the ML and UAI communities, and to foster new collaborations between interested researchers.

Most new developments that are the focus of this workshop concern efficient (in many cases, linear-time) algorithms for theoretically optimal inference procedures that were previously thought not to be efficiently solvable. It is therefore hoped that the workshop will inspire original practical applications of MDL in machine learning domains. Development of such applications recently became a lot easier, because of the new (2007) book on MDL by P. Grünwald [1], which provides the first comprehensive overview of the field, as well as in-depth discussions of how it relates to other approaches such as Bayesian inference. Remarkably, the originator of MDL, J. Rissanen, also published a new monograph in 2007; and a Festschrift in Honor of Rissanen's 75th birthday was presented to him in May 2008.

Bibliography

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

Contents

MDL Tutorial

Peter Grünwald

We give a self-contained tutorial on the Minimum Description Length (MDL) approach to modeling, learning and prediction. We focus on the recent (post 1995) formulations of MDL, which can be quite different from the older methods that are often still called 'MDL' in the machine learning and UAI communities.

In its modern guise, MDL is based on the concept of a 'universal model'. We explain this concept at length. We show that previous versions of MDL (based on so-called two-part codes), Bayesian model selection and predictive validation (a variation of cross-validation) can all be interpreted as approximations to model selection based on 'universal models'. Modern MDL prescribes the use of a certain 'optimal' universal model, the so-called 'normalized maximum likelihood model' or 'Shtarkov distribution'. This is related to (yet different from) Bayesian model selection with non-informative priors. It leads to a penalization of 'complex' models that can be given an intuitive differential-geometric interpretation. Roughly speaking, the complexity of a parametric model is directly related to the number of distinguishable probability distributions that it contains. We also discuss some recent extensions such as the 'luckiness principle', which can be used if the Shtarkov distribution is undefined, and the 'switch distribution', which allows for a resolution of the AIC-BIC dilemma.

The talk assumes no prior knowledge of information theory. The menu is as follows:

1. Codes and Probability Distributions
2. Universal Coding
3. The Bayes, 2-part and Normalized Maximum LIkelihood Universal Model
4. MDL Model Selection
5. Relation to Bayes factor model selection and Cross-Validation
6. The Luckiness Principle, The Switch Distribution

Efficient Computation of NML for Bayesian Networks

Petri Myllymäki
Department of Computer Science & Helsinki Institute for Information Technology
P.O. Box 68, FI-00014 University of Helsinki, Finland

Abstract

Bayesian networks are parametric models for multidimensional domains exhibiting complex dependencies between the dimensions (domain variables). A central problem in learning such models is how to regularize the number of parameters; in other words, how to determine which dependencies are significant and which are not. The *normalized maximum likelihood (NML)* distribution or code offers an information-theoretic solution to this problem. Unfortunately, computing it for arbitrary Bayesian network models appears to be computationally infeasible, but we show how it can be computed efficiently for certain restricted type of Bayesian network models.

1 Normalized Maximum Likelihood

Let

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

be a 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 $\mathbf{x}_{:,1}, \ldots, \mathbf{x}_{:,m}$.

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, a universal model $\hat{p}(x^n)$ 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 [19, 22]

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

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 depending only on the sample size n:

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

For some model classes, the normalizing factor is finite only if the range $\mathcal{X}^n$ of the data is restricted, see e.g. [19, 20, 2]. For discrete models, the normalizing constant, $C_{\mathcal{M}}(n)$, is given by a sum over all data matrices of size $m \times n$:

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

2 Bayesian Networks

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 increasing node index, i.e., $\mathrm{Pa}_j \subseteq \{1, \ldots, j-1\}$. If this is not the case, the columns in the data, and the corresponding nodes in the graph, can be simply relabeled, which does not change the resulting model. Figure 1 gives an example.

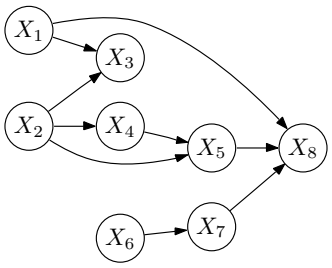


Figure 1: An example of a directed acyclic graph (DAG). The parents of node X_8 are $\{X_1, X_5, X_7\}$. The descendants of X_4 are $\{X_5, X_8\}$.

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

$$\theta_{j|\mathrm{Pa}_j}(r, \mathbf{s}) = p(x_{i,j} = r \mid \mathrm{pa}_{i,j} = \mathbf{s} \; ; \; \theta_{j|\mathrm{Pa}_j}) \quad , \quad 1 \leq i \leq n, 1 \leq j \leq m \; ,$$

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. 1; 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:

$$p(x^n \; ; \; \theta) = \prod_{j=1}^{m} p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \; ; \; \theta_{j|\mathrm{Pa}_j}) \; . \tag{2}$$

This type of probabilistic graphical models are called Bayesian networks [18]. Factorization (2) entails a set of conditional independencies, characterized by so called Markov properties, see [16]. 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.

3 NML for Bayesian Networks

The NML distribution based on (2) and a fixed Bayesian network graph structure $\mathcal{G}$ is given by

$$p_{\mathrm{NML}}(x^n \,;\, \mathcal{G}) = \frac{\prod_{j=1}^{m} p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \,;\, \hat{\theta}(x^n))}{C_{\mathcal{G}}(n)} \;\;, \tag{3}$$

where

$$C_{\mathcal{G}}(n) = \sum_{x^n} \prod_{j=1}^{m} p(\mathbf{x}_{:,j} \mid \mathrm{Pa}_j \,;\, \hat{\theta}(x^n)) \;\;. \tag{4}$$

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

$$\hat{\theta}_{j|\mathrm{Pa}_j}(r, \mathbf{s}) = \frac{\big|\{i \;:\; x_{i,j} = r, \mathrm{pa}_{i,j} = \mathbf{s}\}\big|}{\big|\{i' \;:\; \mathrm{pa}_{i',j} = \mathbf{s}\}\big|} \;\;, \tag{5}$$

where $|S|$ denotes the cardinality of set S. However, direct summing over all possible data matrices is not tractable except in toy problems where n and m are both very small.

For a single (independent) multinomial variable with K values, the normalizing constant can be computed in quadratic time using the recursion [7, 11]:

$$\mathcal{C}_K(n) = \sum_{r_1+r_2=0}^{n} \frac{n!}{r_1! r_2!} \left(\frac{r_1}{n}\right)^{r_1} \left(\frac{r_2}{n}\right)^{r_2} \cdot \mathcal{C}_{K^*}(r_1) \cdot \mathcal{C}_{K-K^*}(r_2), \tag{6}$$

which holds for all $K^* = 1, \ldots, K-1$. A straightforward algorithm based on this formula can be used to compute $\mathcal{C}_K(n)$ in time $\mathcal{O}\left(n^2 \log K\right)$. In [5, 9] the quadratic-time algorithm was further improved to $\mathcal{O}\left(n \log n \log K\right)$ by writing (6) as a convolution-type sum and then using the Fast Fourier Transform algorithm. However, the relevance of this result is unclear due to severe numerical instability problems it easily produces in practice. Moreover, although these results have succeeded in removing the exponentiality of the computation of the multinomial NML, they are still superlinear with respect to n. In [10] a linear-time algorithm based on the mathematical technique of generating functions was derived for the problem. In this paper it was shown how the properties of the so-called *Cayley's tree function* [4, 1] can be used to prove the following remarkably simple recurrence formula:

$$\mathcal{C}_{K+2}(n) = \mathcal{C}_{K+1}(n) + \frac{n}{K} \cdot \mathcal{C}_K(n). \tag{7}$$

It is now straightforward to write an $\mathcal{O}\left(n + K\right)$ time algorithm for computing the multinomial NML based on this result. The algorithm is also very easy to implement and does not suffer from any numerical instability problems.

The one-dimensional single multinomial case is of course not adequate for many real-world situations, where data is typically multi-dimensional and involves complex dependencies between the domain variables, but it is a useful building block that can be exploited with more complex Bayesian networks. An example of a domain where the multinomial NML can be directly applied is histogram density estimation, as demonstrated in [10].

In [11], a quadratic-time algorithm for computing the NML for a specific Bayesian network structure, usually called the Naive Bayes, was derived. In this case the Bayesian network forms a single-layer tree where one of the variables is the root, an the other variables form the leaves. This model family has been very successful in practice in mixture modeling [14], clustering of data [11], case-based reasoning [12], classification [3, 13] and data visualization [8]. The time complexity of the algorithm is $\mathcal{O}\left(n^2 + L\right)$, where L denotes the number of values of the root variable. This result was further improved in [17] to $\mathcal{O}\left(n^2\right)$. For more complex Bayesian network structures, we have been able to derive an algorithm

which runs in polynomial time with respect to the the number of values of the leave nodes, but is exponential with respect to the number of values of the non-leave nodes [15, 23]. For Bayesian networks of arbitrary complexity, it appears that the problem of computing the NML is not feasible [6]. However, recently developed new variants [21] of the standard NML offer an alternative, computationally efficient information-theoretic approach for regularizing Bayesian network models.

Acknowledgments. This work was supported in part by the Academy of Finland under the project Civi and by the Finnish Funding Agency for Technology and Innovation under the projects Kukot and PMMA. In addition, this work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence.

References

[1] R.M. Corless, G.H. Gonnet, D.E.G. Hare, D.J. Jeffrey, and D.E. Knuth. On the Lambert W function. *Advances in Computational Mathematics*, 5:329–359, 1996.

[2] S. de Rooij and P. Grünwald. An empirical study of minimum description length model selection with infinite parametric complexity. *Journal of Mathematical Psychology*, 50(2):180–192, 2006.

[3] P. Grünwald, P. Kontkanen, P. Myllymäki, T. Silander, and H. Tirri. Minimum encoding approaches for predictive modeling. In G. Cooper and S. Moral, editors, *Proceedings of the 14th International Conference on Uncertainty in Artificial Intelligence (UAI'98)*, pages 183–192, Madison, WI, July 1998. Morgan Kaufmann Publishers, San Francisco, CA.

[4] D.E. Knuth and B. Pittel. A recurrence related to trees. *Proceedings of the American Mathematical Society*, 105(2):335–349, 1989.

[5] M. Koivisto. *Sum-Product Algorithms for the Analysis of Genetic Risks*. PhD thesis, Report A-2004-1, Department of Computer Science, University of Helsinki, 2004.

[6] M. Koivisto. Parent assignment is hard for the MDL, AIC, and NML costs. In *Proceedings of The 19th Annual Conference on Learning Theory (COLT 2006)*, Lecture Notes in Computer Science 4005, pages 289–303. Springer, 2006.

[7] P. Kontkanen, W. Buntine, P. Myllymäki, J. Rissanen, and H. Tirri. Efficient computation of stochastic complexity. In C. Bishop and B. Frey, editors, *Proceedings of the Ninth International Conference on Artificial Intelligence and Statistics*, pages 233–238. Society for Artificial Intelligence and Statistics, 2003.

[8] P. Kontkanen, J. Lahtinen, P. Myllymäki, T. Silander, and H. Tirri. Supervised model-based visualization of high-dimensional data. *Intelligent Data Analysis*, 4:213–227, 2000.

[9] P. Kontkanen and P. Myllymäki. A fast normalized maximum likelihood algorithm for multinomial data. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI-05)*, 2005.

[10] P. Kontkanen and P. Myllymäki. A linear-time algorithm for computing the multinomial stochastic complexity. *Information Processing Letters*, 103(6):227–233, 2007.

[11] P. Kontkanen, P. Myllymäki, W. Buntine, J. Rissanen, and H. Tirri. An MDL framework for data clustering. In P. Grünwald, I.J. Myung, and M. Pitt, editors, *Advances in Minimum Description Length: Theory and Applications*. The MIT Press, 2006.

[12] P. Kontkanen, P. Myllymäki, T. Silander, and H. Tirri. On Bayesian case matching. In B. Smyth and P. Cunningham, editors, *Advances in Case-Based Reasoning, Proceedings of the 4th European Workshop (EWCBR-98)*, volume 1488 of *Lecture Notes in Artificial Intelligence*, pages 13–24. Springer-Verlag, 1998.

[13] P. Kontkanen, P. Myllymäki, T. Silander, H. Tirri, and P. Grünwald. On predictive distributions and Bayesian networks. *Statistics and Computing*, 10:39–54, 2000.

[14] P. Kontkanen, P. Myllymäki, and H. Tirri. Constructing Bayesian finite mixture models by the EM algorithm. Technical Report NC-TR-97-003, ESPRIT Working Group on Neural and Computational Learning (NeuroCOLT), 1996.

[15] P. Kontkanen, H. Wettig, and P. Myllymäki. NML computation algorithms for tree-structured multinomial Bayesian networks. *EURASIP Journal on Bioinformatics and Systems Biology*, 2007.

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

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

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

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

[20] J. Rissanen. MDL denoising. *IEEE Transactions on Information Theory*, 46(7):2537–2543, 2000.

[21] T. Roos, T. Silander, P. Kontkanen, and Myllymäki P. Bayesian network structure learning using factorized NML universal models. In *Information Theory and Applications Workshop*, San Diego, CA, January 2008.

[22] Yu.M. Shtarkov. Universal sequential coding of single messages. *Problems of Information Transmission*, 23:3–17, 1987.

[23] 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.

Tracking the Best Predicting Model

Steven de Rooij*

Abstract

According to standard MDL and Bayesian model selection, we should (roughly) prefer the model that minimises overall prediction error. But if the goal is to predict well, it may well depend on the sample size which model is most useful to predict the next outcome. By re-interpreting the Bayesian prediction strategies associated with the models as "experts", we can use the various algorithms for "expert tracking" to improve model selection for prediction without introducing a substantial computational overhead.

1 Model Selection Preliminaries

A *model* $\mathcal{M} = \{P_\theta | \theta \in \Theta\}$ is a set of probability distributions. *Model selection* is choosing the "most useful" model based on the available observations $x^n := x_1, \ldots, x_n \in \mathcal{X}^n$. For simplicity, we consider only model selection criteria that satisfy Dawids *weak prequential principle* [1, 2]. That is, models are considered "useful" if we can use them to construct *prediction strategies* that give high probability to the data, or, equivalently, achieve low accumulated prediction error, where prediction error is measured using logarithmic loss. More discussion about how our results relate to model selection for other applications, such as truth finding, can be found in [4]. To further simplify the presentation, we assume that $\mathcal{X}$ is countable, we identify probability distributions with their defining mass functions, and we treat the sample size n as a given rather than considering random processes.

As the most important special case, we consider Bayes factor model selection, where prior distributions $w_1, \ldots, w_k$ are defined on the parameter spaces $\Theta_1, \ldots, \Theta_K$ of each of the models. By "integrating out" the parameter we obtain, for each model $\mathcal{M}_k$, an associated marginal distribution:

$$P_k(x^n) := \int_{\theta \in \Theta_k} P_\theta(x^n) w_k(\theta) \mathrm{d}\theta. \tag{1}$$

By subsequently defining a prior distribution W on the models, we can then use Bayes' rule to compute the posterior odds

$$\frac{P(\mathcal{M}_i | x^n)}{P(\mathcal{M}_j | x^n)} = \frac{W(i)}{W(j)} \cdot \frac{P_i(x^n)}{P_j(x^n)},$$

in other words, the posterior odds are the prior odds multiplied by the probability ratio of the data, which is called the "Bayes factor".

We now take a step back and use the chain rule for conditional probability to rewrite (1) as

$$P_k(x^n) = P_k(x_1) \cdot P_k(x_2 | x^1) \cdot \ldots \cdot P_k(x_n | x^{n-1}),$$

to obtain a prediction strategy. Thus, Bayes factor model selection satisfies the weak prequential principle, and it is an example of the model selection criteria we consider.

*Based on joint work with Tim van Erven, Wouter Koolen and Peter Grünwald

2 Example: First vs Second Order Markov Chains

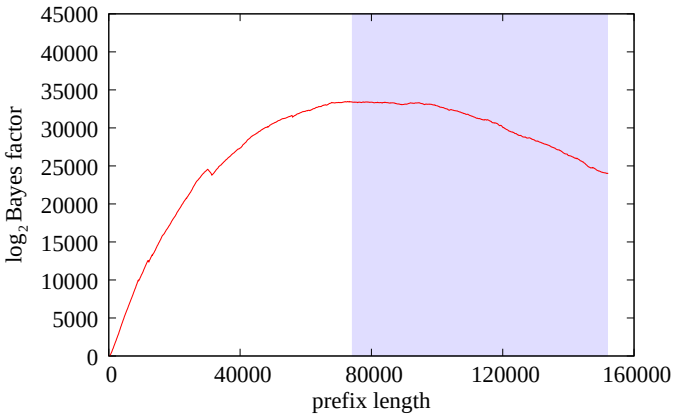
We give a concrete, simple example of Bayes factor model selection. Let $\mathcal{M}_1$ and $\mathcal{M}_2$ be the sets of all first and second order Markov chains on the 8-bit ASCII set $\mathcal{X}$, $|\mathcal{X}| = 256$. The models are parameterised by their transition probabilities. Now let P_1 and P_2 be corresponding Bayesian prediction strategies based on uniform priors $w_1(\theta) = 1$ and $w_2(\theta) = 1$. We also use a uniform prior $W(1) = W(2) = \frac{1}{2}$ on the models. Finally let x^n be the sequence of ASCII symbols that constitute Alice in Wonderland, which has $n = 152089$. We can now calculate

$$\frac{P(\mathcal{M}_1|x^n)}{P(\mathcal{M}_2|x^n)} = \frac{P_1(x^n)}{P_2(x^n)} = \frac{2^{-569147}}{2^{-593132}} = 2^{23985}.$$

Thus, Bayes factor model selection tells us that the odds are overwhelmingly in favour of the first order Markov model. This suggests that we should also expect P_1 to issue better predictions, i.e. if Carroll were to rise from the grave and write an additional chapter to his beloved story, we might expect that P_1 assigns higher probability to, and accumulates less loss on, that new chapter.

This assumption turns out to be false, certainly in this example. The reason is that the incurred loss for P_1 and P_2 is not evenly distributed over the entire sample, which means that even though P_1 may have accumulated less loss overall, it may still be the case that P_2 is making better predictions *at the current sample size*. This becomes very clear if we look at the $\log_2$ of the Bayes factor as a function of the length of the prefix of the novel.

From the graph it is clear that around prefix length 78,000, the two prediction strategies perform more or less equally well, since the Bayes factor hardly changes there. Beyond sample size 78,000, the strategy based on the second order Markov chain model accumulates less loss, causing the Bayes factor to decrease. However P_1 has acquired so much evidence in its favour, that it will take many outcomes before P_2 can finally catch up in terms of accumulated prediction error. Only then will P_2 be preferred by Bayes factor model selection. We call this the *catch-up phenomenon*.

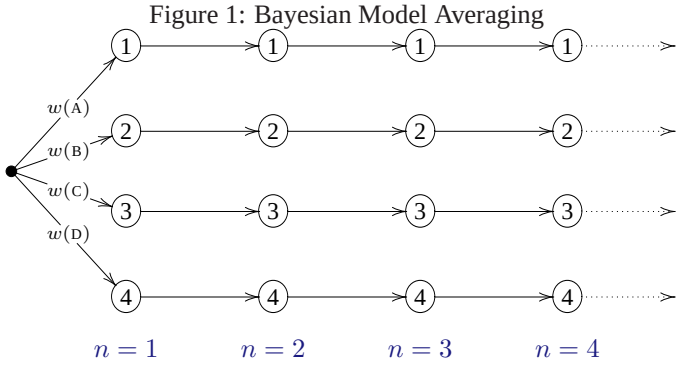


To put this example in perspective, note that we are *not* trying to suggest that either the models or the priors we used are reasonable. We used this extremely naive example only for simplicity and because it *illustrates* the phenomenon we are interested in so well. One may ask if the phenomenon would still occur if we had used better models. The answer is yes: even if the models are chosen carefully so that one of the considered models is "true", i.e. it contains the distribution from which the data were sampled, then it may still be the case that, at lower sample sizes, the prediction strategies associated with other, simpler models may be much more effective, so that the catch-up phenomenon will still occur. Furthermore, the processes we encounter in practice are often so complex that even the best models we can come up with are naive, and we are forced to use uninformative priors. One example is the nonparametric setting, see [4]. We would still like to make the best predictions we can under those circumstances!

3 Expert Tracking

To improve predictive performance when the catch-up phenomenon occurs, we would like to figure out which prediction strategy issues the best predictions, not just overall, but *at each sample size*. For in-

Figure 1: Bayesian Model Averaging



stance, in the Alice example we would want to switch from prediction according to P_1 to prediction according to P_2 around sample size 78,000 rather than never.

It turns out that algorithms to do just this already exist. Namely, we may think of the prediction strategies $P_1, \ldots, P_K$ associated with the models as "experts", which is just another word for an algorithm that issues predictions given a sequence of past observations. We will now describe some known algorithms for prediction advice that are useful in this context.

As explained in [5], many algorithms for prediction with expert advice can be implemented by forward propagation on hidden Markov models (HMMs). Bayesian Model Averaging (BMA) is one of the simplest. It mixes the expert predictions according to their posterior weights as follows:

$$P(x_{n+1}|x^n) = \sum_{k=1}^{K} P_k(x_{n+1}|x^n) w(k|x^n). \tag{2}$$

Figure 1 shows the corresponding HMM for four experts labeled $1, \ldots, 4$. It can be interpreted as a description of a prior distribution, not on the experts, but on *sequences* of experts. Namely, the prior probability that expert k is used at sample size n is the sum of the weights of all paths from the starting state to the state(s) associated with expert k at sample size n. The weight of a path is the product of its transition probabilities. The HMM in Figure 1 contains only one path to each expert and that path visits no other experts. Thus, only expert sequences that contain exactly one expert receive nonzero prior probability: *switching* between experts is not catered for. This inability to switch between experts needs to be addressed in order to alleviate the catch-up problem.

A second simple algorithm for prediction with expert advice goes to the other extreme: here it is not only possible to use different experts at different sample sizes, but which expert is used at sample size $n+1$ does not even *depend* on which expert is used at sample size n! The corresponding prediction strategy is

$$P(x_{n+1}|x^n) = \sum_{k=1}^{K} P_k(x_{n+1}|x^n) w(k). \tag{3}$$

In their groundbreaking paper "Tracking the Best Expert" [3], Herbster and Warmuth interpolate between these two extreme approaches: rather than never or always, they switch to a different expert with fixed probability α, as in Figure 3.[1] Note that, as before, forward propagation on this HMM only needs to maintain K weights, and requires total running time proportional to the number of experts K and the

[1] Actually, unlike FixedShare, the HMM in Figure 3 allows switching to the same expert. However, it can be made to simulate FixedShare by using a slightly lower value for α.

Figure 2: Elementwise Mixture

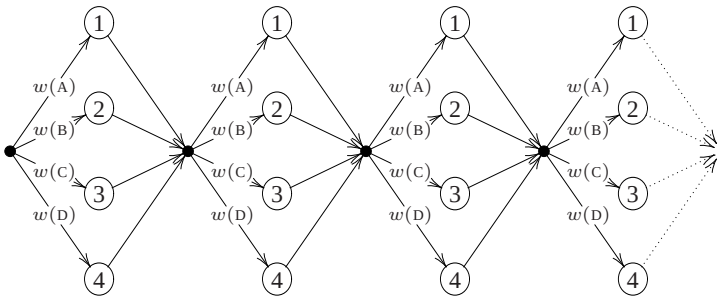
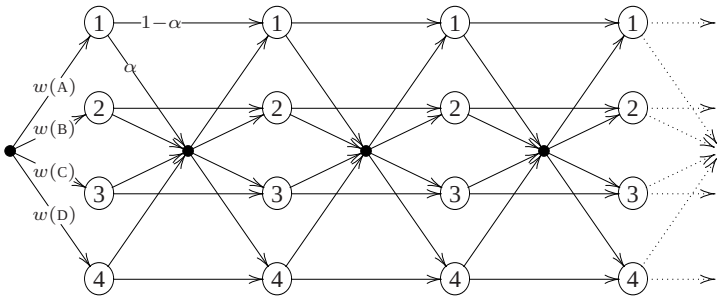


Figure 3: Fixed-Share



sample size n. The corresponding prediction strategy is in between (2) and (3):

$$P(x_{n+1}|x^n) = \sum_{k=1}^{K} P_k(x_{n+1}|x^n)((1-\alpha)P(K_n = k|x^n) + \alpha w(k)). \tag{4}$$

Herbster and Warmuth compare the logarithmic loss incurred by Fixed-Share to that incurred by any partition of the data into m blocks, where any expert is used within each block. They show that

$$\log \frac{P_{\text{m-part}}(x^n)}{P_{\text{fs}}(x^n|\alpha)} \leq (n-1)H(\alpha) + \log K + (m-1)\log(K-1),$$

provided that the parameter α is optimally tuned to $(m-1)/(n-1)$. This result can be applied directly to our Alice in Wonderland example: ideally we would partition the data into $m = 2$ blocks, with the split appearing somewhere around sample size 78,000. The Fixed-Share bound tells us that, compared to this optimal partition, the logarithmic loss using the Fixed-Share algorithm is *at most* $(n-1)H(\frac{1}{n-1})+1 \leq 17$ bits higher. This overhead is negligible compared to the gain, which is of the order of 9,000 bits, as can be read from the log Bayes factor graph. Namely, compared to using P_1 for the entire book, we gain the difference between the height of the graph at index 78,000 (around 33,000 bits) and at full sample size (around 24,000 bits).

The Fixed-Share algorithm does require tuning of the switching rate alpha. However, by letting the probability of the switching transitions decrease as a function of the sample size, it is possible to do away with the parameter α at only a moderate cost in terms of the performance guarantee. More information about other expert tracking algorithms in HMM format is given in [5].

References

[1] A.P. Dawid. Present position and potential developments: Some personal views, statistical theory, the prequential approach. *Journal of the Royal Statistical Society, Series A*, 147(2):278–292, 1984.

[2] A.P. Dawid. Prequential data analysis. In M. Ghosh and P.K. Pathak, editors, *Current Issues in Statistical Inference: Essays in Honor of D. Basu*, Lecture Notes-Monograph Series, pages 113–126. Institute of Mathematical Statistics, 1992.

[3] M. Herbster and M.K. Warmuth. Tracking the best expert. *Machine Learning*, 32:151–178, 1998.

[4] T. van Erven, P.D. Grünwald, and S. de Rooij. Catching up faster in Bayesian model selection and model averaging. In *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, 2008.

[5] W.Koolen and S. de Rooij. Expert automata for efficient tracking. In *Proceedings of the 21st Annual Conference on Computational Learning Theory (COLT)*, 2008.

Extensions to MDL denoising

Janne Ojanen
Helsinki University of Technology TKK
Department of Biomedical Engineering and Computational Science
P.O.Box 9203 (Tekniikantie 14), FI-02015 TKK, Finland
jiojanen@lce.hut.fi

Jukka Heikkonen
European Commission - Joint Research Centre
Institute for the Protection and Security of the Citizen (IPSC)
G04 Maritime Affairs (Fishreg Sector) TP 051, Via Fermi 1, I-21020 Ispra (VA), Italy
jukka.heikkonen@jrc.it

Abstract

The minimum description length principle in wavelet denoising can be extended from the standard linear-quadratic setting in several ways. We describe briefly three extensions: soft thresholding, histogram modeling and a multicomponent approach. The MDL hard thresholding approach based on the normalized maximum likelihood universal modeling can be extended to include soft thresholding shrinkage, which can be considered to give better results in some applications. In MDL histogram denoising approach the assumptions of the parametric density models for the data can be relaxed. The informative and noise components of the data are modeled with equal bin width histograms. The method can cope with different noise distributions. In multicomponent approach more than one non-noise components are included in the model, because it is possible that in addition to the random noise there may be other disturbing signal elements, or that the informative signal is comprised of several different components which we may want to observe, separate or remove. In these cases adding informative components in the model may result result in better performance than in the NML denoising approach.

1 Introduction

The observed data is thought to be corrupted by additive noise, $y^n = x^n + \epsilon^n$, where the noise term ϵ^n is often assumed to be comprised of i.i.d. Gaussians. Given the orthonormal regression matrix $\mathbf{W}$ he discrete wavelet transform (DWT) of the noisy data is defined as $c^n = \mathbf{W}^T y^n$. The aim of wavelet denoising is to obtain modified coefficients $\hat{c}^n$ representing the informative part in the data. In MDL setting wavelet denoising is seen as a model selection task. The linear regression model can be rewritten as a density function $f(y^n|c^n_\gamma, \sigma^2, \gamma) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left\{-\frac{1}{2\sigma^2}||y^n - \mathbf{W}c^n_\gamma||\right\}$, where the structure index γ defines which rows of the regressor matrix are included in the model, or equivalenty, which elements of c^n_γ are non-zero. We may now define the NML density function, compute the well-known maximum likelihood estimates for parameters and calculate the normalizing factor by a renormalization scheme

discussed in Rissanen [1], and the result is a denoising criterion

$$\frac{k}{2}\ln\left(\frac{1}{k}\sum_{i\in\gamma}c_i^2\right)+\frac{n-k}{2}\ln\left(\frac{1}{n-k}\sum_{j\notin\gamma}c_j^2\right)+\frac{1}{2}\ln k(n-k)+L(\gamma) \tag{1}$$

approximating the stochastic complexity. The selection of γ and the resulting coefficient vector c_γ^n are obtained by minimizing the criterion. Furthermore, the criterion is shown to be minimized by the k coefficients with largest magnitudes. For the code length for the model class a code length function $L(\gamma)=\ln\binom{n}{k}$ is recommended in [2].

2 MDL soft thresholding

An extension to wavelet denoising is to include soft thresholding shrinkage [3]. In essence, soft thresholding the observed wavelet coefficients with a threshold parameter λ gives two subsets indexed by γ_1 and γ_2: $\hat{c}_{\gamma_1}=(\hat{c}_{\gamma_1(1)},\ldots,\hat{c}_{\gamma_1(k)})$ corresponding to the shrinked k 'informative' coefficients $\hat{c}_i=\mathrm{sign}(c_i)(|c_i|-\lambda)$, and $\tilde{c}_{\gamma_2}=(\tilde{c}_{\gamma_2(1)},\ldots,\tilde{c}_{\gamma_2(n-k)})$ containing the $n-k$ unmodified 'noise' coefficients for which $\mathrm{sign}(c_i)(|c_i|-\lambda)<0$. A useful analogy is to think the process as data transmission over a channel. The sender must transmit enough information over a channel to the receiver so that the receiver is capable of reconstructing the original data from the transmitted signal. In this case we transmit, with as short a code length as possible, k soft thresholded coefficients $\hat{c}_{\gamma_1}$ and $n-k$ noise coefficients $\tilde{c}_{\gamma_2}$, so that when λ (which also must be transmitted) is known the receiver is able to reconstruct the original data.

The code length for the wavelet coefficients is obtained by encoding the subsets $\hat{c}_{\gamma_1}$ and $\tilde{c}_{\gamma_2}$ with separate NML codes $L_{\mathrm{NML}}(\hat{c}_{\gamma_1}|\gamma_1)$ and $L_{\mathrm{NML}}(\tilde{c}_{\gamma_2}|\gamma_2)$, respectively. For computing the NML code length for any sequence, see [4]. The code length of the model class, $L(\gamma_1,\gamma_2,\lambda)$, is also required for describing the parameter of the shrinkage function as well as the index sets γ_1 and γ_2. The code length may be further divided into $L(\gamma_1,\gamma_2,\lambda)=L(\gamma_1,\gamma_2|\lambda)+L(\lambda)$, where $L(\gamma_1,\gamma_2|\lambda)=\ln\binom{n}{k}$ gives the code length for choosing the k coefficients into γ_1 out of a total of n coefficients when λ is fixed. $L(\lambda)$ is required to describe the threshold parameter value. However, $L(\lambda)$ may be considered to be a constant that can be ignored in the final criterion. Finally, the encoding is performed by a two-part encoding where the total code length is given by the sum $L_{\mathrm{NML}}(\hat{c}_{\gamma_1}|\gamma_1)+L_{\mathrm{NML}}(\tilde{c}_{\gamma_2}|\gamma_2)+L(\gamma_1,\gamma_2,\lambda)$. Applying the Stirling's approximation to Gamma functions and ignoring all terms constant with respect to k gives the criterion for choosing the optimal parameter λ,

$$\min_\lambda\left[\frac{k}{2}\ln\left(\frac{1}{k}\sum_{i=1}^{k}\hat{c}_{\gamma_1(i)}^2\right)+\frac{n-k}{2}\ln\left(\frac{1}{n-k}\sum_{i=1}^{n-k}\tilde{c}_{\gamma_2(i)}^2\right)+\frac{1}{2}\ln k(n-k)+L(\gamma)\right]. \tag{2}$$

The criterion (2) is almost identical to the original MDL denoising criterion (1): the difference is in the first term, where in the soft thresholding criterion there are shrinked wavelet coefficient values instead of the originals.

3 MDL histogram denoising

The NML approach is restricted to the quadratic-linear case in which noise is assumed to follow Gaussian distribution. We obtain another denoising criterion by employing histogram models. The main idea is to model the wavelet coefficients representing the denoised signal, $\hat{c}^n$, by an equal bin width histogram at each resolution level of the wavelet transform, and the coefficients representing the noise, $\tilde{c}^n$ by a single equal bin width histogram. Minimization of the total code length yields the optimal way of dividing the coefficients into ones representing informative signal and noise. For information on how to compute the stochastic complexity for the data string given the number of bins in the fixed bin width histogram, see [5, 6].

The key parts of the MDL-histo denoising algorithm discussed in more detail in [7] are summarized as follows:

1. Obtain the set of wavelet coefficients $c^n = c_1^{n_1}, \ldots, c_r^{n_r}$ through the r-level wavelet transform.
2. Recursively on resolution levels $i = 1, \ldots, r$ fit a an m-bin histogram H_i to the coefficients $c_i^{n_i}$ and select a tentative collection of bins S_i, with the number of chosen bins $m_i = |S_i|$. Denote by $n_{i,(j)}$ the number of points falling in the bin of H_i having index (j). The bins in S_i contain k_i retained coefficients. The retained and residual coefficients at level i are written as two strings $\hat{c}_i^{n_i}$ and $\tilde{c}_i^{n_i}$, respectively.
3. Fit a histogram with M bins to the residual coefficients $\tilde{c}^n = \tilde{c}_1^{n_1}, \ldots, \tilde{c}_i^{n_i}, c_{i+1}^{n_{i+1}}, \ldots, c_r^{n_r}$ where the first i residual strings are obtained by setting the already retained coefficients to zero.
4. Find the optimal S_i by minimizing the criterion

$$
\begin{aligned}
\min_{S_i, M} \Bigg\{ & \log_2 \binom{n_i}{n_{i,(1)}, \ldots, n_{i,(m_i)}, (n_i - k_i)} + \\
& \log_2 \binom{n_i + m_i + 1}{n_i} + \log_2 \binom{n - \sum_{j=1}^{i-1} \hat{k}_j - k_i}{\nu_1, \ldots, \nu_M} \\
& + \log_2 \binom{n - \sum_{j=1}^{i-1} \hat{k}_j - k_i + M}{M} + k_i \log_2 \left(\frac{R}{m} \right) \\
& + k_i \log_2 \left(\frac{M}{R_i} \right) + \left(\sum_{j=1}^{i-1} \hat{k}_j \right) \log_2 \left(\frac{M}{R_i} \right) \\
& - (n-1) \log_2 M + 2 \log_2 \log_2 M \\
& + n \log_2 R_i + \log_2 R_i + 2 \log_2 \log_2 R_i \Bigg\}, \quad (3)
\end{aligned}
$$

   where ν_j is the number of coefficients falling into the jth bin of the M-bin histogram fitted to the residual string $\tilde{c}^n$, R is the range of wavelet coefficients, R_i are the levelwise ranges of the coefficients and $\sum_{j=1}^{i-1} \hat{k}_j$ denotes the number of retained coefficients in the so far optimized sets $S_j, j < i$. For the first level $i = 1$ this sum is zero.

The denoised signal results from the inverse transform of the sequence of retained coefficients $\hat{c} = \hat{c}^n = \hat{c}_1^{n_1}, \ldots, \hat{c}_r^{n_r}$.

4 Multicomponent denoising

It is possible that in addition to the random noise there may be other disturbing signal elements, or that the informative signal is comprised of several different components which we may want to observe, separate

or remove. With more than one informative component in the noisy measured data a multicomponent approach may result in better performance than the original MDL denoising method [8].

Roos et al. [9, 10, 2] have shown that a criterion similar to the renormalization result can be obtained by a different derivation, details of which can be found in [10, 2]. In short, they define a model for the wavelet coefficients, in which each coefficient is distributed according to a zero-mean Gaussian density with variance σ_I^2 if it belongs to the set of informative coefficients indexed by γ, or according to a zero-mean Gaussian density with variance σ_N^2 if it represents noise, with the restriction $\sigma_I^2 \geq \sigma_N^2$. Again, the optimal denoising result is given by the γ minimizing the normalized maximum likelihood code length of the data given the model class defined by γ.

This approach may be extended by using m Gaussian components and specifying the restriction for their variance parameters, $\sigma_1^2 \geq \ldots \geq \sigma_m^2$. The NML code length for this model can be calculated in a manner following the derivation in [10] for the two-component denoising criterion. The derivation turns out to be straightforward since the normalizing integral factors into m parts, each depending only on the coefficients determined by the respective index set γ_i. We obtain a criterion

$$\begin{aligned} \sum_{i=1}^{m} \left(\frac{k_i}{2} \ln \frac{1}{k_i} \sum_{j \in \gamma_i} c_j^2 + \frac{1}{2} \ln k_i \right) \\ + L(\gamma_1, \ldots, \gamma_m) + m \log\log \frac{\sigma_{\max}^2}{\sigma_{\min}^2} + \text{const} \,, \end{aligned} \tag{4}$$

where const refers to terms constant with respect to the index sets and m, and $\sigma_{\max}^2$ and $\sigma_{\min}^2$ are hyperparameters for the maximum and minimum variance, respectively. The last two terms can be ignored if we wish to find the optimal m-component result. However, if we want to compare the results for two approaches with different number of components, for example $m_1 = 3$ and $m_2 = 4$, we cannot remove the term involving the hyperparameters as it affects the code length.

References

[1] J. Rissanen. MDL denoising. *IEEE Transactions on Information Theory*, 46(7):2537–2543, 2000.

[2] T. Roos. *Statistical and Information-Theoretic Methods for Data-Analysis*. PhD thesis, University of Helsinki, 2007.

[3] J. Ojanen and J. Heikkonen. MDL and wavelet denoising with soft thresholding. *Submitted to 2008 Workshop on Information Theoretic Methods in Science and Engineering*, 2008.

[4] J. Rissanen, editor. *Information and Complexity in Statistical Modeling*. Springer, 2007.

[5] P. Hall and E.J. Hannan. On stochastic complexity and nonparametric density estimation. *Biometrika*, 75(4):705–714, December 1988.

[6] J. Rissanen, T.P. Speed, and B. Yu. Density estimation by stochastic complexity. *IEEE Transactions on Information Theory*, 38(2):315–323, March 1992.

[7] V. Kumar, J. Heikkonen, J. Rissanen, and K. Kaski. Minimum description length denoising with histogram models. *IEEE Transactions on Signal Processing*, 54(8):2922–2928, August 2006.

[8] J. Ojanen, J. Heikkonen, and K. Kaski. Towards the multicomponent MDL denoising. In *Festschrift in Honor of Jorma Rissanen on the Occasion of his 75th Birthday*. Tampere University Press, 2008.

[9] T. Roos, P. Myllymäki, and H. Tirri. On the behavior of MDL denoising. In Robert G. Cowell and Zoubin Ghahramani, editors, *AISTATS05*, pages 309–316. Society for Artificial Intelligence and Statistics, 2005.

[10] T. Roos, P. Myllymäki, and J. Rissanen. MDL denoising revisited. *Submitted for publication*, 2006. Preprint available at: `http://www.arxiv.org/abs/cs.IT/0609138`.

Sequential and Factorized NML models

Tomi Silander **Teemu Roos** **Petri Myllymäki**
Helsinki Institute for Information Technology HIIT

1 INTRODUCTION

Bayesian networks are among most popular model classes for discrete vector-valued i.i.d data. Currently the most popular model selection criterion for Bayesian networks follows Bayesian paradigm. However, this method has recently been reported to be very sensitive to the choice of prior hyper-parameters [1]. On the other hand, the general model selection criteria, AIC [2] and BIC [3], are derived through asymptotics and their behavior is suboptimal for small sample sizes.

This extended abstract is based on an unpublished manuscript [4] in which we introduce a new effective scoring criterion for learning Bayesian network structures, the factorized normalized maximum likelihood (fNML). This score features no tunable parameters thus avoiding the sensitivity problems of Bayesian scores. It also has a probabilistic interpretation which yields a natural way to use the selected model for predicting future data.

2 BAYESIAN NETWORKS

Bayesian network defines a joint probability distribution for an n-dimensional data vector $X = (X_1, \ldots, X_n)$, where each X_i may have r_i different values which, without loss of generality, can be denoted as $\{1, \ldots, r_i\}$.

2.1 Model class

A Bayesian network consists of a directed acyclic graph (DAG) G and a set of conditional probability distributions. We specify the DAG with a vector $G = (G_1, \ldots, G_n)$ of parent sets, so that $G_i \subset \{X_1, \ldots, X_n\}$ denotes the parents of variable X_i, i.e., the variables from which there is an arc to X_i. Each parent set G_i has q_i ($q_i = \prod_{X_p \in G_i} r_p$) possible values that are the possible value combinations of the variables belonging to G_i. We assume an enumeration of these values and denote the fact that G_i holds the j^{th} value combination simply by $G_i = j$.

The conditional probability distributions $P(X_i \mid G_i)$ are determined by a set of parameters, Θ, via the equation

$$P(X_i = k \mid G_i = j, \Theta) = \theta_{ijk}.$$

We denote the set of parameters associated with variable X_i by Θ_i. Given a Bayesian network (G, Θ) the joint distribution can be factorized as

$$P(x \mid G, \Theta) = \prod_{i=1}^{n} P(x_i \mid G_i, \Theta_i) = \prod_{i=1}^{n} \theta_{iG_i x_i}. \quad (1)$$

2.2 Data

To learn the Bayesian network structures, we assume data D of N i.i.d instantiations of the vector X, i.e., an $N \times n$ data matrix without missing values. We select columns of the data matrix D by subscripting it with a corresponding variable index or a variable set.

Since the rows D are assumed to be i.i.d, the probability of a data matrix can be calculated by just taking the product of the row probabilities. Combining equal terms yields

$$P(D \mid G, \Theta) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}}, \tag{2}$$

where N_{ijk} denotes number of rows in which $X_i = k$ and its parents contain the j^{th} value combination.

For a given structure G, the maximum likelihood parameters are simply the relative frequencies found in the data: $\hat{\theta}_{ijk} = \frac{N_{ijk}}{\sum_{k'} N_{ijk'}}$. Setting parameters $\hat{\theta}_{ijk}$ to their maximum likelihood values for data D, gives the maximized likelihood $P(X \mid G, \hat{\Theta}(D))$. In the following, we denote the value $P(D \mid G, \hat{\Theta}(D))$ by $\hat{P}(D \mid G)$[1].

3 Model selection

The number of possible Bayesian network models for n variables is super exponential, and the model selection task has been shown to be NP-hard for practically all model selection criteria such as AIC, BIC, and marginal likelihood [5]. However, all popular Bayesian network selection criteria $S(G, D)$ feature a convenient *decomposability*

$$\textsc{Score}(G, D) = \sum_{i=1}^{n} S(D_i, D_{G_i}) \tag{3}$$

that makes implementing a heuristic search for models easier [6].

Many popular scoring functions avoid overfitting by balancing the fit to the data and the complexity of the model. A common form of this idea can be expressed as

$$\textsc{Score}(G, D) = \log \hat{P}(D \mid G) - \Delta(D, G), \tag{4}$$

where $\Delta(D, G)$ is a complexity penalty. For example, $\Delta^{\mathrm{BIC}} = \sum_i \frac{q_i(r_i-1)}{2} \log N$, and $\Delta^{\mathrm{AIC}} = \sum_i q_i(r_i - 1)$.

3.1 Bayesian Dirichlet scores

The current state-of-the-art is to use marginal likelihood scoring criterion

$$S_{\mathrm{BD}}(D_i, D_{G_i}, \bar{\alpha}) = \log \int_{\theta_i} P(D_i | D_{G_i}, \theta_i) W(\theta_i | \alpha_i) d\theta_i. \tag{5}$$

The most convenient form of this, the Bayesian Dirichlet (BD) score, uses conjugate priors in which parameter vectors Θ_{ij} are assumed independent of each other and distributed by Dirichlet distributions so that

$$W(\theta_i \mid \alpha_i) = \prod_{j=1}^{q_i} P(\theta_{ij} \mid \alpha_{ij*}), \tag{6}$$

in which $\theta_{ij} \sim Dir(\alpha_{ij1}, \ldots, \alpha_{ijr_i})$. With a choice of $\alpha_{ijk} = \frac{\alpha}{q_i r_i}$ we get a family of BDeu scores popular for giving equal scores for different Bayesian network structures that encode same independence

[1] We often drop the dependency on G from the notation when it is clear from the context.

assumptions. The BDeu score depends only on single parameter α, but recent studies show that model selection is very sensitive to it.

For predictive purposes it is natural to parameterize the model learned with the BD-score by expected parameter values

$$\theta_{ijk}^{BD} = \frac{N_{ijk} + \alpha_{ijk}}{\sum_{k'=1}^{r_i}[N_{ijk'} + \alpha_{ijk'}]}. \tag{7}$$

4 FACTORIZED NML

The factorized normalized maximum likelihood (fNML) score is based on the normalized maximum likelihood (NML) distribution [7, 8]

$$P_{\mathrm{NML}}(D \mid \mathcal{M}) = \frac{\hat{P}(D \mid \mathcal{M})}{\sum_{D'} \hat{P}(D' \mid \mathcal{M})}, \tag{8}$$

where the normalization is over all data sets D' of a fixed size N. The log of the normalizing factor is called the *parametric complexity* or the *regret*. Evaluation of the regret is often hard due to the exponential number of terms in the sum. We propose a decomposable factorized normalized maximum likelihood criterion with a local score

$$S_{\mathrm{NML}}(D_i, D_{G_i}) = \log P_{\mathrm{NML}}(D_i \mid D_{G_i}) = \log \left(\frac{\hat{P}(D_i \mid D_{G_i})}{\sum_{D_i'} \hat{P}(D_i' \mid D_{G_i})} \right),$$

where the normalizing sum goes over all the possible D_i-column vectors of length N, i.e., $D_i' \in \{1, \ldots, r_i\}^N$. Using recently discovered methods for calculating the regret for a single r-ary multinomial variable [9] the fNML-criterion can be calculated as efficiently as other decomposable scores.

For predictive purposes its is natural to parameterize the model learned with the fNML-score by predictive conditional NML parameters [10]

$$\theta_{ijk} = \frac{e(N_{ijk})(N_{ijk} + 1)}{\sum_{k'=1}^{r_i} e(N_{ijk'})(N_{ijk'} + 1)}, \tag{9}$$

where $e(n) = (\frac{n+1}{n})^n$.

Empirical tests with real data sets indicate that the fNML selection criterion performs very well in a code length sense when compared with the state of the art BDeu criterion. The predictive capabilities of the Bayesian and fNML approaches are currently under investigation.

References

[1] T. Silander, P. Kontkanen, and P. Myllymäki, "On sensitivity of the MAP Bayesian network structure to the equivalent sample size parameter," in *Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence*, R. Parr and L. van der Gaag, Eds. 2007, pp. 360–367, AUAI Press.

[2] H. Akaike, "Information theory and an extension of the maximum likelihood principle," in *Proceedings of the Second International Symposium on Information Theory*, B.N. Petrox and F. Caski, Eds., Budapest, 1973, pp. 267–281, Akademiai Kiado.

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

[4] T. Silander, T. Roos, P. Kontkanen, and Myllymäki, "Factorized normalized maximum likelihood criterion for learning bayesian network structures," Submitted for PGM08, 2008.

[5] D.M. Chickering, "Learning Bayesian networks is NP-Complete," in *Learning from Data: Artificial Intelligence and Statistics V*, D. Fisher and H. Lenz, Eds., pp. 121–130. Springer-Verlag, 1996.

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

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

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

[9] 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.

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

Generalization Theory of Two-part Code MDL Estimator

Tong Zhang
Department of Statistics
Rutgers University

I will present a finite-sample generalization analysis of two-part code MDL estimator. This method selects a model that minimizes the sum of the model description length plus the data description length given the model. It can be shown that under various conditions, optimal rate of convergence can be achieved through an extended family of two-part code MDL that over-penalize the model description length.

As an example, we apply MDL to learning sparse linear representations when the system dimension is much larger than the number of training examples. This is a problem that has attracted considerable attention in recent years. The generalization performance of a two-part code MDL estimator is calculated based on our theory, and it compares favorably to other methods such as 1-norm regularization.

Segmentation of DNA sequences using Normalized Maximum Likelihood models for uncovering gene duplications

Ioan Tăbuş
Department of Signal Processing
Tampere University of Technology
P.O. Box 553, FIN-33101 Tampere, Finland
email: ioan.tabus@tut.fi
web: www.cs.tut.fi/∼tabus

Abstract

The normalized maximum likelihood (NML) model [2]-[4] for a class of Markov sources [6] was recently used for the compression of full genomes, obtaining for the human genome the best existing compression results [1]. We show that one of the underlying biological features that the compression algorithm implicitly uncovers is the existence of approximate gene duplication. We proposed a refined method based on the same NML models for the segmentation of DNA sequences for uncovering gene duplications [5]. Several analysis tasks in genomic sequences involve preliminary segmentation or clustering of the data, which can be performed by a number of techniques, based on various similarity measures. Here we review and further pursue the application of MDL techniques for genomic sequence analysis. The process of sequence matching will be used for solving the problem of uncovering gene duplications with the help of a preliminary segmentation of a complex DNA locus, known to have evolved through a series of duplications.

References

[1] G. Korodi, I. Tabus, "Normalized maximum likelihood model of order-1 for the compression of DNA sequences", in Proc. IEEE Data Compression Conference, DCC'07, pp:33 - 42, Snowbird, 27-29 March 2007.

[2] J. Rissanen, "Fisher information and stochastic complexity", *IEEE Transactions on Information Theory*, vol. IT-42, pp. 40–47, Jan. 1996.

[3] J. Rissanen, "Strong optimality of the normalized ML models as universal codes and information in data", vol. IT-47 (5), pp. 1712-1717, 2001.

[4] Y.M. Shtarkov, "Universal sequential coding of single messages". Translated from *Problems of Information Transmission*, Vol. 23, No. 3, 3–17, July-September 1987.

[5] I. Tabus, Y. Yang, J. Astola, "Universal models with memory for genomic sequence analysis", 3rd International Symposium on Communications, Control and Signal Processing, ISCCSP 2008, March 1214, St. Julians, Malta, 2008.

[6] I. Tabus, G. Korodi, "Genome compression using normalized maximum likelihood models for constrained Markov sources", *IEEE Information Theory Workshop*, Porto, Portugal, May 5-9, 2008.

Information Consistency of Nonparametric Gaussian Process Methods

Matthias W. Seeger

Joint work with S. Kakade and D. Foster [1].

We present information consistency results for nonparametric sequential prediction with Gaussian processes. The connection to nonparametric MDL is through the prequential approach, as detailed in Grünwald's 2007 book, Sect. 13.5. Our proof technique is elementary, making use of a convex duality previously useful to obtain PAC-Bayesian bounds. We also obtain precise information consistency rates for a wide range of kernels and input distributions, using kernel eigenvalue asymptotics. In all these cases, the linear expert space is an infinite-dimensional function space, but still very reasonable rates are obtained.

References

[1] M. Seeger, S. Kakade, and D. Foster. Information consistency of nonparametric Gaussian process methods. *IEEE Transactions on Information Theory*, 54(5):2376–2382, 2008.