PREFACE

The Ninth Workshop on Information Theoretic Methods in Science and Engineering (WITMSE 2016) took place on September 19–21, 2016, in Helsinki, Finland. The workshop was organized jointly by the Department of Computer Science of the University of Helsinki and the Helsinki Institute for Information Technology HIIT.

The WITMSE series started in 2008 and has continued annually at locations in Tampere (2008–2009), Helsinki (2011), Amsterdam (2012), Tokyo (2013), Honolulu (2014) and Copenhagen (2015). As the title of the workshop suggests, WITMSE seeks speakers from a variety of disciplines with emphasis on both theory and applications of information and coding theory with special interest in modeling. Since the beginning our plan has been, and still is, to keep the number of the participants small and to ensure the highest possible quality, which has been accomplished by inviting distinguished scholars as speakers.

The workshop programme included 22 invited talks and three plenary talks that were given by Andrew Barron, John Shawe-Taylor, and Wojciech Szpankowski.

Outside the technical sessions, the program included a ferry trip to Suomenlinna sea fortress where a welcoming reception was held on Monday 19th September. On the following day, we enjoyed a banquet dinner at the restaurant Savu, located on Tervasaari island.

We would like to thank all the participants to the workshop, and the Helsinki Doctoral Programme in Computer Science (DoCS) for financial support.

We hope to see many of you again next year.

November 7, 2016
San Jose and Helsinki
Workshop Co-Chairs
Jorma Rissanen,
Janne Leppä-aho,
Teemu Roos,
& Petri Myllymäki
Contents

Preface ......................................................... 3

Plenary Talks

Andrew Barron: Probability Models for Normalized Maximum Likelihood Calculation . 6
John Shawe-Taylor: Learning to Detect an Immune Response from T-Cell Sequences . 7
Wojciech Szpankowski: Analytic Pattern Matching: From DNA to Twitter ............ 8

Invited Papers

David R. Bickel: A Bayesian Approach to Informing Decision Makers: Comparisons to
Minimizing Relative Entropy .................................... 9
Marc Boullé, Fabrice Clérot, Carine Hue: Revisiting Enumerative Two-part Crude
MDL for Bernoulli and Multinomial Distributions .............................. 12
Jukka Corander: 100 Years After William Bateson - What Can We Learn About Epistasis
by Today's Statistical Machine Learning? ...................................... 16
Ivo Grosse: The Entropic Hourglass of Animal and Plant Embryogenesis ........ 18
Peter D. Grünwald: Safe Testing .................................... 19
Jean Honorio: On the Sample Complexity of Learning Sparse Graphical Games .... 21
Samuel Kaski: Multiple Output Regression with Latent Noise ..................... 22
Michael Mahoney: Sub-sampled Second-order Newton Methods for Large-scale Machine
Learning ..................................................................... 23
Kohei Miyaguchi, Shin Matsushima, Kenji Yamanishi: Stochastic Complexity for
Sparse Modeling .................................................... 24
Paavo Nevalainen, Juuso Suomi, Jukka Heikkinen: Heuristically Targeted Minimum
Description Length Test for Stone Detection from Public Point Cloud Data .... 26
Dmitri Pavlichin: Chained Kullback-Leibler Divergences ........................... 30
Boris Ryabko, Nadezhda Savina: Two-Faced Processes and Pseudorandom Number Generators ......................................................... 31

Tomi Silander: Bayesian Network Structure Learning with a Quotient Normalized Maximum Likelihood Criterion .................................................. 32

Joe Suzuki: Jeffreys’ and BDeu Priors for Model Selection .................................................. 36

Ioan Tabus: Sparse Modelling Using Information Theoretic Structure Selection .................. 40

Kazuho Watanabe: Rate-distortion Theoretic Views of Learning Problems .................. 41

Kenji Yamanishi: Stochastic Complexity for Latent Variable Models .................. 45
PROBABILITY MODELS FOR NORMALIZED MAXIMUM LIKELIHOOD CALCULATION

Andrew Barron
Yale University

ABSTRACT

We show how a probability model in which the counts are independent with probability function \( p(k) \) proportional to the Stirling ratio \( k^k e^{-k} / k! \) produces exactly the Normalized Maximum Likelihood (NML) Shtarkov distribution upon conditioning on the total counts. This Stirling ratio distribution leads to clean and fast algorithms for computing the conditional distributions needed for arithmetic coding of data according to the NML. Also, it allows for simple computation of the Shtarkov constant analogous to what was obtained by Petri Kontkanen and Petri Myllymaki (2005). Bayes mixtures with possibly negative weights (Barron, Roos, Watanabe 2014) provide an alternative method of computing the joint and conditional Shtarkov distribution. However, the Stirling ratio method is superior when the alphabet size is large. This is joint work with Xiao Grace Yang.
LEARNING TO DETECT AN IMMUNE RESPONSE FROM T-CELL SEQUENCES

John Shawe-Taylor

University College London

ABSTRACT

Adaptive immunity can be viewed as a computational system that learns to respond to different pathogens. The response results in changes in the frequencies of T-Cells. The problem of detecting this response requires detecting these changes. The information is distributed across many different T-cells and an approach using Fisher kernels to highlight the relevant features is presented. Links between Fisher kernels and string kernels from finite state automata suggest extensions to Fisher kernel feature sets that allow biologically significant features to be extracted efficiently.
ANALYTIC PATTERN MATCHING: FROM DNA TO TWITTER

Wojciech Szpankowski
Purdue University

ABSTRACT
Repeated patterns and related phenomena in words are known to play a central role in many facets of computer science, telecommunications, coding, data compression, data mining, and molecular biology. One of the most fundamental questions arising in such studies is the frequency of pattern occurrences in a given string known as the text. Applications of these results include gene finding in biology, executing and analyzing tree-like protocols for multiaccess systems, discovering repeated strings in Lempel-Ziv schemes and other data compression algorithms, evaluating string complexity and its randomness, synchronization codes, user searching in wireless communications, and detecting the signatures of an attacker in intrusion detection. In this talk after a brief motivation, we review several pattern matching problems (e.g., exact string matching, constrained pattern matching, generalized pattern matching, and subsequence pattern matching), and then we discuss a few applications (e.g., spike trains of neuronal data, Google search, Lempel-Ziv’77 and Lempel-Ziv’78 data compression schemes, and string complexity used in Twitter classification). Finally, we illustrate our approach to solve these problems using tools of analytic combinatorics, which we discuss in some depth. We also present several open problems.
A BAYESIAN APPROACH TO INFORMING DECISION MAKERS: COMPARISONS TO MINIMIZING RELATIVE ENTROPY

David R. Bickel

University of Ottawa

1. AVAILABILITY

The working paper (Bickel, 2016b) and slides (Bickel, 2016a) corresponding to this extended abstract are available from http://www.davidbickel.com.

2. EXTENDED ABSTRACT

Let \( \Theta \) denote the set of possible values of \( \theta \), the parameter of interest. Considering an observed sample \( x \) of size \( n \), the probability mass or probability density of \( x \) is \( f_\theta (x) \) for all \( \theta \in \Theta \). Every non-empty, measurable \( H \subset \Theta \) corresponds to the hypothesis that \( \theta \in H \). That hypothesis is simple if \( H \) has a single member or composite if it has at least two members.

The observation \( x \) is modeled as a realization of a random variable \( X \) of distribution \( f_\theta \), written as \( X \sim f_\theta \). The prior probabilities \( P(\theta \in H) \) and \( P(\theta \notin H) \) are related to the posterior probabilities \( P(\theta \in H|X = x) \) and \( P(\theta \notin H|X = x) \) by Bayes’s theorem:

\[
\frac{P(\theta \in H|X = x)}{P(\theta \notin H|X = x)} = \frac{f(x|\theta \in H) P(\theta \in H)}{f(x|\theta \notin H) P(\theta \notin H)} \tag{1}
\]

where \( f(x|\theta \in H) \) and \( f(x|\theta \notin H) \) are probability densities under the hypotheses that \( \theta \in H \) and \( \theta \notin H \), respectively. The three ratios in equation (1), from left to right, are called the posterior odds, the Bayes factor, and the prior odds. Whereas prior odds and posterior odds are directly applicable for betting on hypotheses before and after considering the observation \( x \), the Bayes factor quantifies the strength of evidence supporting the hypotheses that \( \theta \in H \) and over the hypothesis that \( \theta \notin H \) (contra Vieland and Seok, 2016). While the Bayes factor is often called a “likelihood ratio” (LR) in forensic science (Curran, 2016), the more narrowly defined LR is \( f_{\theta_1}(x) / f_{\theta_0}(x) \), the strength of evidence supporting the simple hypothesis that \( \theta = \theta_1 \) over the simple hypothesis that \( \theta = \theta_0 \) (e.g., Blume, 2011; Hodge et al., 2011; Zhang and Zhang, 2013; Rohde, 2014, Ch. 17; Bandy-
opadhyay et al., 2016, Ch. 2). That LR is a special case of the Bayes factor for prior and posterior probabilities conditional on $\theta \in \{\theta_0, \theta_1\}$.

For various reasons, the prior probabilities (Troffaes and de Cooman, 2014) or the Bayes factor (Berger and Slooten, 2016) may not be known precisely, resulting in imprecision in the posterior probabilities. However, decision makers may need a scientist to report a single value for a posterior probability or Bayes factor, a value that can be used to make decisions about whether to reject a hypothesis (Ommen et al., 2016). The choice of which feasible distribution or Bayes factor to report depends on how it will be used to inform decisions, that is, on the utility functions or loss functions of the decision makers.

When that future use is unknown, the distribution of loss functions that is both scale-invariant and reciprocal-invariant leads to reporting a geometric mean of odds (Bickel, 2016b, §2) or a geometric mean of Bayes factors (Bickel, 2016b, §3) in order to minimize the degree to which an adversary can discredit the report. Those geometric means may either be unweighted and thus epistemic or weighted according to a degree of caution as determined by considerations specific to an application. That observes the distinction between scientific inference and practical decision making (cf. Fisher, 1973). The approach is extended in Bickel (2016b, §4) to the problem of reporting a distribution such as a posterior distribution. When the hypotheses about which decisions will be made are also unknown, the reported distribution minimizes a distance from the distribution selected by an adversary. The reported posterior distribution is the same whether such minimization is applied to the priors or to the posteriors. Analogously, in the presence of a reference measure such as an initial prior, minimizing that distance commutes with conditioning on the observed data much more generally than does minimizing relative entropy.

3. ACKNOWLEDGMENTS

This research was partially supported by the Natural Sciences and Engineering Research Council of Canada (RGPIN/356018-2009) and by the Faculty of Medicine of the University of Ottawa.

4. REFERENCES


Bickel, D. R., 2016a. A Bayesian approach to informing decision makers: Comparisons to minimizing relative entropy. Slides presented at the Ninth Workshop on Information Theoretic Meth-
ods in Science and Engineering.
URL http://wp.me/p5eoC-hh


Vieland, V. J., Seok, S.-C., 2016. Statistical evidence measured on a properly calibrated scale for multinomial hypothesis comparisons. Entropy 18 (4), 114.

ABSTRACT
We leverage the Minimum Description Length (MDL) principle as a model selection technique for Bernoulli distributions and compare several types of MDL codes. We focus on the enumerative two-part crude MDL code, suggest a Bayesian interpretation for finite size data samples, and exhibit a strong connection with the NML approach. We obtain surprising impacts on the estimation of the model complexity together with superior compression performance. This is then generalized to the case of the multinomial distributions. Both the theoretical analysis and the experimental comparisons suggest that one might use the enumerative code rather than NML in practice, for Bernoulli and multinomial distributions.

This paper is an extended abstract of the research report [1], which contains all the detailed analysis, proofs and extensive experiments.

1. INTRODUCTION

Model selection is a key problem in statistics and data mining, and the MDL approaches [2] to model selection have been extensively studied in the literature [3], with successful applications in many practical problems. Simple models such as Bernoulli or mainly multinomial distributions are important because they are easier to analyze theoretically and useful in many applications. For example, the multinomial distribution has been used as a building block in more complex models, such as naive Bayes classifiers, Bayesian networks, decision trees or coclustering models. These models involve up to thousands of multinomials blocks, some of them with potentially very large numbers of occurrences and outcomes. For example in [4], half a billion call detail records (occurrences) are distributed on one million coclusters (outcomes). These various and numerous applications critically rely on the use of effective and efficient MDL code lengths to get a robust and accurate summary of the data.

The MDL approaches come with several flavors, ranging from theoretical but not computable to practical but sub-optimal. Ideal MDL [5] relies on the Kolmogorov complexity, that is the ability of compressing data using a computer program. However, it suffers from large constants depending on the description method used and cannot be computed, not even approximated in the case of two-part codes [6]. Practical MDL leverages description methods that are less expressive than general-purpose computer languages. It has been employed to retrieve the best model given the data in case of families of parametrized statistical distributions. Crude MDL is a basic MDL approach with appealing simplicity. In two-part crude MDL, you just have to encode the model parameters and the data given the parameter, with a focus on the code length only. However, crude MDL suffers from arbitrary coding choices. Modern MDL relies on universal coding resulting in Refined MDL [3], with much stronger foundations and interesting theoretical properties. In this paper, we investigate the enumerative two-part crude MDL code for the Bernoulli and multinomial models, exhibit a strong connection with the NML approach, with surprising impacts on the estimation of the model complexity and superior compression performance.

The rest of the paper is organized as follows. Section 2 describes a particular two-part crude MDL code based on enumerations and establishes the connection of its parameter coding length with its NML parametric complexity. Section 3 summarizes comparisons between this enumerative MDL code and the standard NML code for Bernoulli distributions. Section 4 presents an extension of the enumerative two-part crude MDL code to multinomial distributions. Finally, Section 5 summarizes this paper.

2. ENUMERATIVE TWO-PART CRUDE MDL

We present the enumerative two-part crude MDL code for Bernoulli distributions, suggest a finite data sample Bayesian interpretation and show a connection with the NML approach. Let us consider the Bernoulli model with \( \theta \in [0, 1] \) in the case of binary sequences \( x^n \in X^n \) of size \( n \). Let \( k(x^n) \) be the number of ones in \( x^n \).

2.1. Enumerative interpretation

The enumerative two-part crude MDL code for Bernoulli distributions has already been proposed in the past literature, under the names of index or enumerative code (see for example [3] Example 10.1 Coding by Giving an Index). First, we enumerate all possible \( \theta = \frac{1}{n} \) parameter values given the sample size \( n \). We then use \( \log(n + 1) \) bits to encode \( \theta \). Second, given \( \hat{\theta}(x^n) = \frac{k(x^n)}{n} \), we enumerate all the \( \binom{n}{k} \) binary sequences with \( k = k(x^n) \) ones and encode the data \( x^n \) using \( \log \binom{n}{k} \) bits. This gives a total code length of
\[ L(\hat{\theta}(x^n), x^n) = \log(n+1) + \log \frac{n!}{k!(n-k)!}. \] (1)

Interestingly, this crude MDL approach results in the same code length as that obtained using Predictive Coding or Mixture Coding with a uniform prior \([3]\). This code has also been studied by \([3]\) (Chapter 10, Section 10.2) under the name Conditional Two-Part Universal Code, which suggests that at least for the Bernoulli model, this code is strictly equivalent to the ordinary two-part code.

2.2. Bayesian interpretation

Let \( \mathcal{M} = \{P_\theta \mid \theta \in [0, 1]\} \) be the class of all Bernoulli distributions. We propose to focus on the family of models \( \mathcal{M}^{(n)} = \{P_\theta \mid \theta = \frac{i}{n}, 0 \leq i \leq n\} \) that are models of description for finite size data samples. \( \mathcal{M}^{(n)} \) is related to the set of all the possible maximum likelihood estimates of \( \theta \) (from \( \mathcal{M} \)) for binary strings of size \( n \). The interest of using \( \mathcal{M}^{(n)} \) is that the number of model parameters is now finite instead of uncountable infinite. Using a uniform prior on the model parameters in \( \mathcal{M}^{(n)} \), we get \( P(\theta = \frac{i}{n}) = 1/|\mathcal{M}^{(n)}| \), leading to \( L(\theta) = \log(n+1) \).

Given \( \theta = \frac{i}{n} \in \mathcal{M}^{(n)} \), we now have to encode the data \( x^n \). If \( k(x^n)/n \neq \theta \), we cannot encode the data and \( P(x^n|\theta) = 0 \). If \( k(x^n)/n = \theta \), the observed data is consistent with the model parameter, and we assume that all the possible observable data are uniformly distributed. The number of binary strings with \( k \) ones is the binomial coefficient \( \binom{n}{k} \). Thus the probability of observing one of them is \( P(x^n|\hat{\theta}(x^n)) = 1/\binom{n}{k} \). We have a discrete likelihood that concentrates the probability mass on binary strings that can be observed given the model parameter. As a result, coding lengths are defined only for strings that are consistent with the model parameter. This gives a total code length of

\[ L(\hat{\theta}(x^n), x^n) = \log(n+1) + \log \frac{n!}{k!(n-k)!}. \] (2)

2.2.1. Generative model for the enumerative Bernoulli distribution

Given a sequence length \( n \) and \( \theta = \frac{i}{n} \in \mathcal{M}^{(n)} \), we can formulate these models as generative models of sequences with exactly \( i \) ones and \( n-i \) zeros. For example, from a sequence of \( n \) zeros, we randomly choose \( i \) times without replacement a zero in the sequence and replace it with a one. For this generative model, we have the following likelihood, as seen previously:

\[ P(x^n|\theta = \frac{i}{n}) = \prod \left( \frac{\theta}{n} \right)^i \left( \frac{1-\theta}{n} \right)^{n-i} \binom{n}{k}(x^n)^{i} \binom{n}{k}(x^n)^{n-i}. \] (3)

For the case of the Bayes mixture model with uniform prior \( w(\theta) = \frac{1}{n+1}, \theta = \frac{i}{n}, 0 \leq i \leq n \), we have

\[ P_{Bayes}(x^n) = \sum_{i=0}^{n} \frac{w(\frac{i}{n})}{n} P(x^n|\theta = \frac{i}{n}), \] (4)

\[ = \frac{1}{n+1} \frac{k(x^n)!}{(n-k(x^n))!} \left( \frac{1}{n} \right)^{k(x^n)} \left( \frac{1}{n} \right)^{n-k(x^n)}. \] (5)

The negative log of this probability actually corresponds to the code length of the enumerative code. Interestingly, the standard Bernoulli model and the enumerative one are related to slightly different generative models, but their Bayes mixture under the uniform prior leads to the same distribution. In Section 2.3, we will see that on the opposite, their normalized maximum likelihood distribution is not the same.

2.2.2. Cardinality of models spaces

Let us consider the union of the \( \mathcal{M}^{(n)} \) models for all the sample sizes:

\[ \mathcal{M}^{(n)} = \bigcup_{n \in \mathcal{N}} \mathcal{M}^{(n)}. \] (6)

Interestingly, \( \mathcal{M}^{(k)} \) is very close to \( \mathcal{M} \), with \( \theta \in \mathcal{R} \) rather than \( \theta \in \mathcal{Q} \). Thus, the number of model parameters in \( \mathcal{M}^{(k)} \) is countable infinite rather than uncountable infinite, which provides a significant simplification.

2.3. NML interpretation

Let us compute the NML parametric complexity of this enumerative code, on the basis of the discrete likelihood presented in Section 2.2. We have

\[ COMP^{(n)}(\theta) = \log \sum_{y^n \in X^n} P_{\hat{\theta}(y^n)}(y^n), \] (7)

\[ = \log \sum_{k=0}^{n} \binom{n}{k} \left( \frac{1}{\binom{n}{k}} \right) \] (8)

\[ = \log(n+1). \] (9)

Interestingly, we find exactly the same complexity term \( \log(n+1) \) as the coding length of the best hypothesis in the enumerative two-part crude MDL code presented in Section 2.1. This shows that the enumerative code is both a two-part and a one-part code. It is parametrization invariant and optimal w.r.t. the NML approach, with minimax regret guarantee. Surprisingly, its parametric complexity is asymptotically twice that of the NML code or the standard BIC regularization term. We further investigate on the comparison between the enumerative and NML codes in next section.

3. CODE COMPARISON FOR THE BERNOULLI DISTRIBUTION

| Code name | \( COMP^{(n)}_{name} \) | \( L_{name}(x^n|\hat{\theta}(x^n)) \) |
|-----------|------------------------|---------------------------------|
| enumerative | \( \log(n+1) \) | \( \log \frac{n!}{k!(n-k)!} \) |
| NML | \( \frac{1}{2} \log \frac{n^2}{2} + o(1) \) | \( \log \frac{n^2}{2} + o(1) \) |

In this section, we compare the standard NML code \([3]\) and enumerative two-part crude MDL code (Section 2) for the Bernoulli distribution. Table 1 reminds the parametric and stochastic complexity of each considered code.

The theoretical and empirical comparison results presented below are a summary of the extended report \([1]\).
3.1. Stochastic complexity term

Let $\delta L \left( x^n \| \hat{\theta}(x^n) \right) = L_{\text{num}} \left( x^n \| \hat{\theta}(x^n) \right) - L_{\text{enum}} \left( x^n \| \hat{\theta}(x^n) \right)$. The stochastic complexity term of the enumerative code is always smaller than that of the NML code for non-degenerated binary strings:

$$\forall n, \forall x^n \in X^n, \ 0 < k(x^n) < n, \delta L \left( x^n \| \hat{\theta}(x^n) \right) > 0. \quad (10)$$

Using the approximation given in [3] (formula 4.36) with the Bernoulli parameter $\theta = \hat{\theta}(x^n)$, we have

$$\delta L \left( x^n \| \hat{\theta}(x^n) \right) = \frac{1}{2} \log(2\pi n \nu \text{var}(\theta)) + O(1/n). \quad (11)$$

The difference of coding length is always positive but not uniform. For $k(x^n) = 0$, $\delta L \left( x^n \| \hat{\theta}(x^n) \right) = 0$. For $k(x^n) \approx n/2$, $\delta L \left( x^n \| \hat{\theta}(x^n) \right) \approx \frac{1}{2} \log(\frac{n}{2})$.

These results demonstrate that the enumerative code provides a better encoding of the data with the help of the model for any binary strings, all the more for strings with equidistributed zeros and ones.

3.2. Parametric complexity term

The parametric complexity term of the enumerative code is always strictly greater than that of the NML code and asymptotically twice it.

$$\forall n > 1, \text{COMP}^{(n)}_{\text{enum}} > \text{COMP}^{(n)}_{\text{NML}}. \quad (12)$$

$$\lim_{n \to \infty} \frac{\text{COMP}^{(n)}_{\text{enum}}}{\text{COMP}^{(n)}_{\text{NML}}} = 2. \quad (13)$$

3.3. Overall code length

Both codes have the same length for two parameter values $\{\theta_{\text{inf}}, \theta_{\text{sup}}\}$, with $\theta_{\text{inf}} \approx 0.114$ and $\theta_{\text{sup}} = 1 - \theta_{\text{inf}}$.

For heavily unbalanced Bernoulli distributions $\theta \in [0, \theta_{\text{inf}}] \cup [\theta_{\text{sup}}, 1]$, the NML code is shorter and for $\theta \in (0, 1]$, $|\theta| \approx \frac{1}{2} \log n - \log \frac{2}{3}$.

For balanced Bernoulli distributions $\theta \in [\theta_{\text{inf}}, \theta_{\text{sup}}]$, the enumerative code is shorter and for $\theta \approx \frac{1}{2}$, $|\theta| \approx \log \frac{n}{2}$.

3.4. Empirical comparisons

Extensive comparisons are reported in [1]. Under the uniform distribution, most binary strings are better compressed with the enumerative code and the average compression is slightly better than using the NML code, with a margin that is asymptotically about $\log \frac{n}{2}$.

In a biased versus fair coin classification experiment, both the NML and enumerative codes are used as classifiers by predicting a bias if they can encode a sequence with a coding length shorter than that of the random code ($n \log 2$), and predicting fair otherwise. Overall, both codes exhibit a similar behavior w.r.t. the coin classification problem, with accuracy increasing from 0.5 for small $n$ to 1 for large $n$, and a slow increase rate for small bias and a fast one for large bias. Except in the tiny samples with $n \leq 20$, the difference of accuracy between the two codes never exceeds around 15%. However, there are some interesting differences. The enumerative code is better at detecting bias while the NML code is better at detecting fair, and the overall accuracy of prediction exhibits a variety of behaviors, with tiny differences. When the bias is small ($\theta_{\text{bias}}$ close from $\frac{1}{2}$), the enumerative code is slightly more accurate in the non-asymptotic case, needing less data to achieve a correct accuracy. When the bias is large ($\theta_{\text{bias}}$ close from 0 or 1), the advantage is this time in favor of the NML code. In all cases, the differences between both codes get tiny for large $n$, in the asymptotic case.

4. THE CASE OF MULTINOMIAL DISTRIBUTION

Let us consider the multinomial model with parameter $\theta = (\theta_1, \ldots, \theta_m)$, $\sum_{j=1}^m \theta_j = 1, \forall j ; \theta_j > 0$, such that $P(x = j) = \theta_j$, in the case of $m$-ary sequences $x^n \in X^n$ of size $n$. For a given sequence $x_n$, $P(x_n) = \prod_{j=1}^m \theta_j^{n_j}$, where $n_j$ is the number of occurrences of outcome $j$ in sequence $x^n$.

4.1. Enumerative two-part crude MDL

Like in the Bernoulli case, the enumerative code for multinomial can be obtained using a two-part crude MDL approach, a Bayesian interpretation or a NML interpretation. We present below the Bayesian interpretation.

Given a sample size $n$, the number of tuples $(n_1, \ldots, n_m)$ such that $\sum_{j=1}^m n_j = n$ is $\binom{n+m-1}{m-1}$. We then encode the multinomial model parameter using a uniform prior

$$P(\theta = \left( \frac{n_1}{n}, \frac{n_2}{n}, \ldots, \frac{n_m}{n} \right)) = 1/\binom{n+m-1}{m-1},$$

leading to $L(\theta) = \log \binom{n+m-1}{m-1}$.

Second, we have to encode the data $x^n$ at best given the $\theta$ parameter. We suggest using a probability distribution for encoding the finite size data sample $x^n$, with the following likelihood. For $\theta \neq \left( \frac{n_1(x^n)}{n}, \frac{n_2(x^n)}{n}, \ldots, \frac{n_m(x^n)}{n} \right)$, we cannot encode the data and $P(x^n|\theta) = 0$. For $\theta = \hat{\theta}(x^n) = \left( \frac{n_1(x^n)}{n}, \frac{n_2(x^n)}{n}, \ldots, \frac{n_m(x^n)}{n} \right)$, the observed data is consistent with the model parameter and we assume that all the possible observable data are uniformly distributed. The number of m-ary strings where the number of occurrences of outcome $j$ is $n_j$ is given by the multinomial coefficient $\frac{\binom{n}{n_1} \binom{n}{n_2} \ldots \binom{n}{n_m}}{n!}$. Thus the probability of observing one particular m-ary string is $P(x^n|\hat{\theta}(x^n)) = 1/\frac{n!}{n_1!n_2!\ldots n_m!}$.

This gives a total code length of

$$L(\hat{\theta}(x^n), x^n) = \log \left( \frac{n + m - 1}{m - 1} \right) + \log \frac{n!}{n_1!n_2!\ldots n_m!}.$$

(14)
4.2. Theoretical and empirical comparisons

The NML code has a parametric complexity that can be either approximated [3, 7] with errors that are hard to quantify in the non-asymptotic case or calculated exactly (e.g. algorithm in $o(n + m)$ [8]) at the expense of computation time. Table 2 presents the parametric and stochastic complexity of each considered code.

Table 2. Parametric and stochastic complexity per code.

| Code name | $\text{COMP}_\text{name}$ | $L_{\text{name}}(x^n|\hat{\theta}(x^n))$ |
|-----------|----------------|--------------------------------|
| enumerative | $\log \left( \frac{n+m-1}{m-1} \right)$ | $\log \left( \frac{n^m}{n_{\hat{\theta}}^{m-1} n_m} \right)$ |
| NML | $\approx \frac{m-1}{2} \log \frac{n}{2\pi}$ | $\log \left( \frac{n^n}{n_1 ... n_m} \right)$ |

In [1], extensive comparisons are reported, regarding the stochastic complexity terms, the parametric complexity terms, the overall code length, the expectation of the code length of all the m-ary sequences under the uniform distribution and the detection of biased dices. Overall, the results are similar to the case of Bernoulli distributions, with differences that increase linearly with the number of parameters.

5. SUMMARY

In this paper, we have revisited the enumerative two-part crude MDL code for the Bernoulli model, which compares favorably with the alternative standard NML code. We have suggested a Bayesian interpretation of the enumerative code, that relies on models for finite size samples and results in a discrete definition of the likelihood of the data given the model parameter. We have shown that the coding length of the model parameter is exactly the same as the model complexity computed by applying the NML formula using the definition of the enumerative maximum likelihood. This means that the enumerative code is both a one-part and two part code, which brings parametrization independence, optimality and simplicity. Surprisingly, the obtained parametric complexity is twice that of the alternative classical NML code or the standard BIC regularization term. The enumerative code has a direct interpretation in terms of two part codes for finite sample data. The model parameter is encoded using a uniform prior w.r.t. the sample size and the data are also encoded using a uniform prior among all the binary strings of given size that can be generated using the model parameter. Experimental comparisons between the enumerative and NML codes show that they are very similar, with small differences only. Under the uniform distribution, the enumerative code compresses most individual sequences slightly better, resulting in a slightly better compression on average. An application to the detection of biased coins demonstrates that the enumerative code has a better sensitivity to biased coins at the expense of more false detections in case of fair coins, but the differences are small and vanish asymptotically.

Extension to the multinominal model is also presented. Using the same approach, we obtain a very simple and interpretable analytic formula for the parametric complexity term, that once again is approximately twice that of the alternative classical NML code or the standard BIC regularization term. The resulting code, both one-part and two-part, is optimal w.r.t. NML approach and parameterization invariant, with a much simpler parametric complexity term. It compresses most strings better than the “classical” NML code with a constant margin and extremely few heavily unbalanced strings with a margin logarithmic in the sample size. Experimental comparisons extend the results obtained with Bernoulli distributions. Both codes are very similar, with small differences that roughly increase linearly with the number of model parameters.

Altogether, the theoretical and experimental results suggest that one might use the enumerative code rather than NML in practice, for Bernoulli and multinominal distributions.

6. REFERENCES


Roughly 100 years ago William Bateson coined the term epistasis, which is today generally considered as a form of interaction between two loci in DNA. Research in epistasis has over the years revealed fascinating biological insights, however, until very recently, progress has been slowed by limited availability of densely sampled population data. Given the advent of latest sequencing technology we are finally facing a possibility to query what experiments nature has performed concerning epistasis. I will present latest research results from a close collaboration with the Pathogen Genomics group at WTSI on how we can advance understanding about epistasis in genomes by exploiting core concepts from statistical physics and statistical machine learning algorithms for ultra-high dimensional models. Our genome-wide epistasis analysis reveals interacting networks of resistance, virulence and core machinery genes in Streptococcus pneumoniae.
PRUNING RULES FOR LEARNING PARSIMONIOUS CONTEXT TREES

Ralf Eggeling

University of Helsinki

ABSTRACT
Summary: We give a novel algorithm for finding a parsimonious context tree (PCT) that best fits a given data set. PCTs extend traditional context trees by allowing context-specific grouping of the states of a context variable, also enabling skipping the variable. However, they gain statistical efficiency at the cost of computational efficiency, as the search space of PCTs is of tremendous size. We propose pruning rules based on efficiently computable score upper bounds with the aim of reducing this search space significantly. While our concrete bounds exploit properties of the BIC score, the ideas apply also to other scoring functions. Empirical results show that our algorithm is typically an order-of-magnitude faster than a recently proposed memory-intensive algorithm, or alternatively, about equally fast but using dramatically less memory. Joint work with Mikko Koivisto.

1. REFERENCES

THE ENTROPIC HOURGLASS OF ANIMAL AND PLANT EMBRYOGENESIS

Ivo Grosse

Martin-Luther-Universität Halle-Wittenberg

ABSTRACT

One surprising observation going back to pioneering works of Karl Ernst von Baer in 1828 and Ernst Haeckel in 1866 is that embryos of different animal and plant species express on average evolutionarily young genes at the beginning of embryogenesis, evolutionarily old genes in mid-embryogenesis, and again evolutionarily young genes at the end of embryogenesis. The origin of this molecular hourglass pattern of animal and plant embryogenesis has remained concealed, but here we find that not only the mean age of expressed genes changes during embryogenesis in an hourglass-like manner, but the whole age distribution of expressed genes changes. Specifically, when studying the entropy of this age distribution as a function of time, we find an hourglass pattern that surprisingly is orders of magnitude more significant than the original hourglass pattern of the mean, which might indicate that the entropic hourglass pattern is more fundamental than, and possibly even the origin of, the original hourglass pattern of animal and plant embryogenesis. Joint work with Hajk-Georg Drost, Alexander Gabel, and Marcel Quint.
Safe Testing

Peter D. Grünwald
CWI and Leiden University
P.O. Box 94079
NL-1090 GB Amsterdam
The Netherlands
E-mail: pdg@cwi.nl

October 31, 2016

We propose a general theory of testing hypotheses based on sequential gambling, where the evidence against the null hypothesis is measured by the amount of money one would win under odds that would be fair if the null hypothesis were true. The idea of such a 'nonnegative Martingale' approach goes back at least to Vovk (1993). We replace the concept of a Martingale test by the somewhat less restrictive safe test — every Martingale test is a safe test but not vice versa. If the null hypothesis $H_0$ is simple and the alternative $H_1$ is composite, then safe testing turns out to be equivalent to MDL testing. Here we work out, for the first time, the case that the null hypothesis is composite as well; that this is possible at all is by no means evident. Importantly, we allow the composite null to be nested ($H_0 \subset H_1$), which allows us to deal with classical problems such as testing whether the mean of a normal with unknown variance is 0, as usually done by the $t$-test. We present two approaches: one with a fairly straightforward implementation which works for nested exponential families; and one that works for general $H_0$ and $H_1$, but does not seem easy to implement in general. The latter approach is based on a new interpretation of the reverse information projection, a concept studied in great detail by Andrew Barron and Jonathan Li (1999), and generalized recently by Grünwald and Mehta (2016) to the GRIP (generalized reverse information projection).

For the simple $H_0$ case, safe testing is equivalent to standard MDL testing, in which one associates $H_1$ with a universal code of choice (such as the NML, Bayesian, prequential plug-in or switch distribution van der Pas and Grünwald (2014)). For composite $H_0$, the safe test still has a data compression interpretation: one can view it as associating also $H_0$ with a universal code. But code is quite different from any of the codes usually employed in ‘standard’ MDL.

The safe testing approach avoids many of the pitfalls of the standard $p$-value and Bayes factor based methods. Most importantly, the result it delivers has a ‘robust $p$-value interpretation’ which remains valid even if, after observing the result, one decides to gather some additional data and do a second test. This is impossible with standard $p$-value-based tests, and it only works with Bayesian tests if one really believes one’s prior, which in practice is rarely the case. As a result, combining evidence from different studies is straightforward. Safe testing can also be interpreted in terms of safe probability (Grünwald, 2016), a generic approach that allows for frequentist guarantees on performance when aspects of models or (Bayesian) priors are incorrect.

The slides of the talk can be found on my webpage, www.grunwald.nl.
References


ON THE SAMPLE COMPLEXITY OF LEARNING SPARSE GRAPHICAL GAMES

Jean Honorio

1 Computer Science, Purdue University, West Lafayette, IN 47907, USA, jhonorio@purdue.edu

ABSTRACT

We study the sample complexity of learning sparse graphical games from purely behavioral data. That is, we assume that we can only observe the players’ joint actions and not their payoffs. We analyze the sufficient and necessary number of samples for the correct recovery of the set of pure-strategy Nash equilibria (PSNE) of the true game. We focus on sparse directed graphs with $n$ nodes and at most $k$ parents per node. We state that if the number of samples is greater than $O(kn \log^2 n)$, then maximum likelihood estimation correctly recovers the PSNE with high probability. The above matches the information-theoretic limits. That is, we state that if the number of samples is less than $\Omega(kn \log^2 n)$, then any conceivable method fails to recover the PSNE with arbitrary probability.
MULTIPLE OUTPUT REGRESSION WITH LATENT NOISE

Samuel Kaski
Aalto University

ABSTRACT
In high-dimensional data, structured noise caused by observed and unobserved factors affecting multiple target variables simultaneously, imposes a serious challenge for modeling, by masking the often weak signal. Therefore, (1) explaining away the structured noise in multiple-output regression is of paramount importance. Additionally, (2) assumptions about the correlation structure of the regression weights are needed. We note that both can be formulated in a natural way in a latent variable model, in which both the interesting signal and the noise are mediated through the same latent factors. Under this assumption, the signal model then borrows strength from the noise model by encouraging similar effects on correlated targets. We introduce a hyperparameter for the latent signal-to-noise ratio which turns out to be important for modelling weak signals, and an ordered infinite-dimensional shrinkage prior that resolves the rotational unidentifiability in reduced-rank regression models. Simulations and prediction experiments with metabolite, gene expression, FMRI measurement, and macroeconomic time series data show that our model equals or exceeds the state-of-the-art performance and, in particular, outperforms the standard approach of assuming independent noise and signal models. Joint work with Jussi Gillberg, Pekka Marttinen, Matti Pirinen, Antti J. Kangas, Pasi Soininen, Mehreen Ali, Aki S. Havulinna, Marjo-Riitta Marjo-Riitta J rvelin and Mika Ala-Korpela.
SUB-SAMPLED SECOND-ORDER NEWTON METHODS FOR LARGE-SCALE MACHINE LEARNING

Michael Mahoney

UC Berkeley

ABSTRACT

A major challenge for large-scale machine learning, and one that will only increase in importance as we develop models that are more and more domain-informed, involves going beyond high-variance first-order optimization methods. Here, we consider the problem of minimizing the sum of a large number of functions over a convex constraint set, a problem that arises in many data analysis and machine learning applications, as well as many more traditional scientific applications. For this class of problems, we establish improved bounds for algorithms that incorporate sub-sampling as a way to improve computational efficiency, while maintaining their original convergence properties. These methods exploit recent results from Randomized Linear Algebra on approximate matrix multiplication. Within the context of second order methods, they provide quantitative convergence results for variants of Newton’s methods, where the Hessian and/or the gradient is uniformly or non-uniformly sub-sampled, under much weaker assumptions than prior work.
STOCHASTIC COMPLEXITY FOR SPARSE MODELING

Kohei Miyaguchi1, Shin Matsushima2 and Kenji Yamanishi3

1Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan,
2kohei.miyaguchi@mist.i.u-tokyo.ac.jp,
3shin.matsushima@mist.i.u-tokyo.ac.jp,
yamanishi@mist.i.u-tokyo.ac.jp.

In this study we consider the problem of identifying the true sparse model underlying the data of interest in view of the minimum description length (MDL) principle. In order to estimate the true model out of an exponentially large number of possible models in a scalable manner, we conduct continuous relaxation on the stochastic complexity, which is a model-selection criterion of the MDL principle. We also derive an efficient method for minimizing RSC in which LASSO estimates are iteratively computed.

1. INTRODUCTION

In sparse modeling, we are motivated to estimate a sparse parameter \( \theta \in \mathbb{R}^d \) involved in a model of data sources, assuming that the true underlying parameter \( \theta^* \) is also sparse. Since the volume of the set of sparse parameters is typically zero by the definition, ordinary methods of parameter estimation such as the maximum likelihood method cannot be straightforwardly applied to the problem of sparse modeling. In this context, we introduce probabilistic interpretation on the parameter

\[
\mathcal{M} = \{ p(X; \theta) \mid \theta \in \Omega \subset \mathbb{R}^d \},
\]

where \( X = (x_1, x_2, \ldots, x_n)^T \in \mathcal{X}^n \) denotes observed data, and the family of its subsets

\[
\mathcal{M}_J = \{ p(X; \theta) \in \mathcal{M} \mid \theta_j = 0, \forall j \notin J \},
\]

indexed by \( J \), the sparsity pattern of the parameter \( \theta \). By selecting a sparsity pattern \( \hat{J} \) with a methodology of model selection, we can obtain a sparse estimate of \( \theta \). However, there is a major challenge in the model selection that the number of candidates \( J \) grows exponentially as the number of dimensionality \( d \) increases. Since the model selection is typically conducted by minimizing some criteria \( C(X; \cdot) \) over all the possible candidates,

\[
\hat{J}(X) \overset{\text{def}}{=} \arg\min_J C(X; J),
\]

it means that direct optimization of the criteria is often unrealistic.

On the other hand, LASSO[1] is widely used for estimating a sparse parameter \( \theta \) without considering an exponentially large number of probabilistic models. It gives sparse estimates with a positive probability by solving the following formulation

\[
\hat{\theta}(X, \lambda) \overset{\text{def}}{=} \arg\min_{\theta \in \Omega} \left\{ \log \frac{1}{p(X; \theta)} + \sum_j \lambda_j |\theta_j| \right\}, \tag{2}
\]

given the hyper-parameter \( \lambda \in \mathbb{R}^d \). Since the hyper-parameter \( \lambda \) is strongly relevant to the resulting estimate \( \hat{\theta}(X, \lambda) \) in consideration of sparse modeling, there been many researches that address the problem of choosing \( \lambda \) appropriately [2, 3, 4]. However, to our best knowledge, there is no established method or criterion for choosing \( \lambda \) such that it induces a consistent estimate of the true sparsity pattern \( J^* \). Note that, although some studies mention the oracle property of the LASSO estimate, e.g. [5], where asymptotic rates of \( \lambda = \lambda_n \) which yield such consistent estimates are given, one cannot determine a good value of \( \lambda \) relative to fixed finite samples \( X \) at hand.

In this study, we consider approximating the stochastic complexity

\[
C(X; J) = -\log \frac{p(X; \hat{\theta}(X, J))}{\int p(Y; \theta(Y, J))d\theta}, \tag{3}
\]

which yields consistent model selection (1) on the basis of the MDL principle[6], so that we can choose the true sparsity pattern \( J^* \) by the way of continuous relaxation. We also derive an efficient algorithm for solving the relaxed problem in which the problem of LASSO (2) is iteratively solved.

2. RELAXATION OF STOCHASTIC COMPLEXITY

The stochastic complexity is a model-selection criterion based on the MDL principle: It is the minimax-optimal prefix code length with respect to the Shtarkov’s minimax regret[7],

\[
C(\cdot, J) = \arg\min_J \max_{L: \text{prefix}} \max_{X, q \in \mathcal{M}} \left\{ L(X) - \log \frac{1}{q(X)} \right\}. \tag{4}
\]

In this sense, the stochastic complexity can be seen as the most universal distribution of the model \( \mathcal{M}_J \) and it
is uniquely determined just by $M_J$. This universality leads to the use of the stochastic complexity as a consistent model-selection criterion [6].

Now we relax the stochastic complexity carefully so as to ease the computational difficulty of its optimization but not to lose its universality. The minimax optimality can be written in the following functional form:

$$C(J) = \arg\min_{L_{\text{prefix}}} \max_{X, \theta \in \Omega} \left\{ L(X) - \log \frac{1}{p(X; \theta)} \prod_j \chi_j(\theta_j^{|\theta_j|}) \right\},$$

using the conventional definition of $0^0 \equiv 1$ and $\log 0 = -\infty$. Here we define $\chi_j(a)$ as the indicator function of set $A$, which takes one if $a \in A$, otherwise zero. Note that $\varphi(J) \equiv (\chi_J(1), \chi_J(2), \ldots, \chi_J(d))^\top$ is a natural vector representation of the model $M_J$. Therefore we have the relaxed stochastic complexity (RSC)

$$\bar{C}(\cdot, v) \equiv \arg\min_{L_{\text{prefix}}} \max_{X, \theta \in \Omega} \left\{ L(X) - \log \frac{1}{p(X; \theta)} \prod_j \theta_j^{|\theta_j|} \right\},$$

by defining intermediate states $v \in [0, 1]^d$ as continuous relaxation of the vector representation $\varphi(J) \in [0, 1]^d$. RSC can be also written in a form similar to the original stochastic complexity,

$$\bar{C}(X; v) = h(X, \lambda) + Z(\lambda),$$

$$h(X, \lambda) \equiv -\log \max_{\theta \in \Omega} p(X; \theta) e^{-\sum\lambda_j^{|\theta_j|}},$$

$$Z(\lambda) \equiv \log \int \max_{\theta \in \Omega} p(Y; \theta) e^{-\sum\lambda_j^{|\theta_j|}} dY,$$

where we define $\lambda \equiv -\log v$ in an element-wise manner. This explicit formulation reveals the close relationship between RSC and LASSO since $h(X, \lambda)$ is equal to the minimized objective value of LASSO (2), and the solution of maximization in $h(X, \lambda)$ is the solution of LASSO, $\hat{\theta}(X, \lambda)$. Therefore RSC can be regraded as a criterion for choosing the hyper-parameter of LASSO.

3. MINIMIZATION OF RELAXED STOCHASTIC COMPLEXITY

The direct computation of the value of RSC is often intractable or demanding due to the normalizing term $Z(\lambda)$. Since the domain of $v$ is continuous, we can utilize the information of gradient of RSC in its minimization in order to avoid computing the objective value: Under mild regularity conditions, by differentiating both sides of (6), we have

$$\frac{\partial}{\partial \lambda} \bar{C}(X; v) = \frac{\partial}{\partial \lambda} h(X, \lambda) + \frac{\partial}{\partial \lambda} Z(\lambda)$$

$$= \tilde{\theta}(X, \lambda) - \mathbb{E}_{q(Y; \lambda)}[\tilde{\theta}(Y, \lambda)],$$

where $q(X; \lambda) \equiv e^{-\bar{C}(X, v)}$ is a probability density function. Here we denote coefficient-wise absolute value of $\theta$ by $|\theta|$. Now we can estimate the minimizer of RSC by iteratively updating $\lambda$ toward the opposite direction from the gradient $\frac{\partial}{\partial \lambda} \bar{C}(X; v)$ with a decreasing step size.

The expectation appearing in the last line can be approximated with the methodology of stochastic gradient descent (SGD), replacing it with a synthetic sample $\tilde{\theta}(X, \lambda)$ where $Y_t \in \mathcal{X}^{\times m}$ is subject to the density $q(Y_t; \lambda)$. The sample $Y_t$ can be efficiently drawn by the Metropolis-Hastings algorithm without computing $Z(\lambda)$. Namely, given a transition probability density function $\pi(Y_t | Y_{t-1})$, we generate a new sample $Y_t$ using the previous sample $Y_{t-1}$ as

$$Y_t = \tilde{Y}_t + (1 - z) Y_{t-1},$$

where $z$ is subject to Bernoulli($\min\{1, \frac{\pi(Y_{t-1} | \tilde{Y}_t)}{\pi(Y_t | \tilde{Y}_t)}\}$).

We can easily compute the ratio of the density $q(X; \lambda)$ since the normalizing factors are canceled out.

4. CONCLUSION

We presented a framework of sparse modeling based on the MDL principle that approximates the stochastic complexity over a continuous domain keeping its meaning as universal code length. We also proposed an efficient method for minimizing RSC employing stochastic gradient descent and Markov chain Monte Carlo sampling.

5. ACKNOWLEDGMENTS

This study was partially supported by JST, CREST.

6. REFERENCES


HEURISTICALLY TARGETED MINIMUM DESCRIPTION LENGTH TEST FOR STONE DETECTION FROM PUBLIC POINT CLOUD DATA

Paavo Nevalainen¹ and Juuso Suom² and Jukka Heikkonen¹

¹Dept. of Information Tech., ²Dept. of Geography and Geology, University of Turku, FI-20014 Turku, FINLAND, ptneva@utu.fi

ABSTRACT

Coarse cross-terrain point clouds are gathered by aerial laser scan (ALS) and dense point clouds by unmanned vehicle (UAV) operation. These two data sources have complementary nature and should be combined for various applications. This paper uses minimum description (MDL) length approach to detect individual stones and their physical dimensions from UAV data. The MDL procedure is spatially targeted by a two-step heuristics: local stoniness likelihood derived from ALS data and the curvature detection on UAV data. Comparison of the performance of MDL principle and a geometric approach, namely mean square error (MSE) minimization is presented. The MDL approach can be applied to cloud point densities \( \rho \geq 3 \text{ m}^{-2} \).

1. INTRODUCTION

The problem of estimating terrain surface stoniness is related to locating automatically potential gravel deposits nearby infrastructure sites in Northern Finland. One can produce a 20 × 20 m raster file about stoniness likelihood [1]. Also, geomorphological models are getting more complex using many micro-topological features, and it seems that the stone size distribution and stone coverage ratio could serve as two new, ubiquitous features.

Publicly available nation-wide aerial laser scan (ALS) 3D point clouds have the ground point density \( \rho = 0.7 \text{ m}^{-2} \), which does not allow the detection of individual stones. At some sites higher ground point densities \( \rho = 42 \text{ m}^{-2} \) are produced by unmanned aerial vehicles (UAV). The minimum description length principle (MDL) is proposed to detect individual stones and their radius from these dense clouds. The MDL process needs a good initial guess of each stone location, and a spatial angle filtering (SAF) algorithm is suggested to produce a list of most likely stone positions at each site with a high stoniness likelihood.

The encoding cost approximation of the point cloud points is based on so called crude MDL [2], where the model and data are encoded separately. A further assumption is made at the proximity of an object about normal distribution of point cloud points in orthogonal direction from the object, and about uniform distribution along the surface of the object. This assumption is common in point cloud research, see e.g. [3].

The Section 2 introduces the crude version of the MDL principle applied to detecting the ground as a planar object and stones as hemispherical objects in the 3D point cloud. A generic “potato” shape is addressed in Section 2.2. Choosing the likely stony areas for further analysis and pinpointing MDL check to likely stone locations is briefly outlined in Section 3. A comparison between geometric stone parameter fitting and MDL, and some early results from a test site with a dense point cloud and known geometry of some stones is presented in Section 4. Conclusions are provided in Section 5.

2. MDL OF VARIOUS POINT SETS

In this treatise a point set \( X \) gets associated to an object \( X \), if the set has an advantageous distribution at the normal of the surface of the object \( X \). One has to compare the cost of encoding a point set \( Q \subset [0, R]^3 \subset \mathbb{R}^3 \), associating points either to a symbolic outliers object \( O \) or to the object \( X \). The first case is assumed to have uniform distribution (even it is known to be too pessimistic). A pixel accuracy \( \epsilon \) and a nominal length \( R \) define the encoding cost \( \phi_3 = 3 \text{lb}(G) \) of one 3D point, where \( G = R/\epsilon \) is the characteristic pixel amount and \( \text{lb}(\cdot) = \log_2(\cdot) \) is an abbreviation of a binary logarithm.

Now, the encoding length of whole cloud assuming all points belonging to an outlier object \( O \), is: \( \Phi(G|O) = |G|\phi_3 \), where \(|G|\) stands for the size of the local point cloud \( G \). Assuming an object \( X \) included to the model, the encoding length becomes:

\[
\Phi(G|O,X) = C|G| + \phi_3|O| + \Phi(X) + \Phi(X|X),
\]

(1)

where \( C = 2 \) is the encoding length to distinguish the point classes. In this case \( Q \) has to be divided to two distinct sets, outliers \( O \) and the set \( X \) : \( Q = O \cup X \). The object parameter encoding length \( \Phi(X) \) and the object-specific point information \( \Phi(X|X) \) are also needed. A summary of objects and their parameter encoding lengths is given in Table 1. The only 2D objects is the line \( L_2 \). The rest are 3D objects: \( L_3, C, P, S \) and \( E \) as a line, a circle, a plane, a hemisphere and an ellipsoid, respectively. A generic smooth object \( M \) (potato) will be introduced in Section 2.2. The shapes \( S, E, M \) are alternative representations of a stone.

The question is, which encoding is shorter: \( \Phi(G|O) \) or \( \Phi(G|X, O) \)? Initially any encoding based on the presence...
of an object $X$ is more costly, but there is a break-even point at $|X| = n_2$. There is also a geometric definition limit $|X| = n_1 < n_2$, at which the object $X$ becomes geometrically defined and where the geometric error of a fit equals zero.

Each point $p \in X$ will be projected on the object surface by $\text{proj}(p,X)$ over an orthogonal distance $d(p,X) = \|p - \text{proj}(p,X)\|_2$. Both components will be encoded separately by tangential and vertical encoding with point-specific costs $\phi_t(\cdot)$ and $\phi_v(\cdot)$, respectively:

$$\Phi(X|X) = \sum_{p \in X} \{\phi_t(\text{proj}(p,X)) + \phi_v(d(p,X))\}. \quad (2)$$

The cost $\phi_v(v)$ of vertical encoding of a single real value $v \sim N(0, \sigma_v^2)$ is the classical Huffman encoding result [3]:

$$\phi_v(v) = \frac{1}{2 \sin^2(\frac{\pi}{2})} \left(\frac{v^2}{\sigma_v^2}\right) + \frac{1}{2} lb(2\pi) \quad (3)$$

where the expected encoding cost $\phi_e$ of vertical distances $v$ follows by choosing $\sigma_v = 1.5 \epsilon$ which leads to $E[v/\sigma_v] = \sqrt{2/\pi}$. The choice is for informal consideration only, and it can be justified by the observed ground height distribution in the dense point cloud.

The tangential part is uniform distributed, and the cost $\phi_t = \mathbb{E}_{p \in X}[\phi_t(\text{proj}(p,X))]$ for expressing one point projection can be derived case by case (see Table 1). Derivation of $\phi_t$ of a line $L_2$ and a D3 plane $P$ have been given in [3]. Other cases are similar, and the derivation is excluded from this presentation due the space considerations. The break-even point cloud size $n_2$ in Table 1 can now be calculated by:

$$\Phi(X|X) = \Phi(X|O)$$

$$\Phi(X) + n_2 (\phi_t + \phi_v) = n_2 \phi_3 \quad (4)$$

A summary of minimum detectable stone radii with sparse and dense data is given in Table 2. One can see that the MDL principle cannot be used for nation-wide sparse data! A rather typical stone with radius $r = 0.6 \text{ m}$ would need a point density $\rho \geq 11/\text{m}^2$ to be detected as a hemispherical object ($\rho \geq 23/\text{m}^2$ for an ellipsoid).

An efficient implementation of stone detection consists of two tasks: to properly initialize an object $X$ (its location and radius) and to improve the initial parameter choice in MDL sense by addressing points to two possible classes: outliers $O$ and object-specific points $X$. We propose a heuristical likelihood for a stone derived from coarse point cloud for the first task. The second task is best done by a random sample consensus (RANSAC) algorithm [4]. Note that actually there are three possible models: $\Phi(G|O)$, $\Phi(G|P,O)$ and $\Phi(G|S,P,O)$, for outliers only, a plane (as the local ground surface) and outliers, and for a stone, a plane and outliers, respectively. The class encoding length $C$ equals 0, 1 and 2 for each three cases. A basic RANSAC needs to be modified for the last case, details are not included in this presentation.

### 2.1. The striped point cloud

The vertical projection of the point cloud is very seldom uniformly distributed. The laser scan process produces stripes, therefore the tangential encoding cost will be affected. The projection $\text{proj}(p,X)$ changes to a projection to a nearest sweep line $L_3$. Lines are at regular intervals and one can associate points to correct line by assuming continuous tangential co-ordinate on subsequential lines or circles. Thus following changes are needed in Table 1:

$$\phi_t = \Phi_3/3 \quad \text{for plane } P \quad \text{and sphere } S \quad (5)$$

Also, one has to encode the indexing of the sweep planes by $\phi_i = lb(R^2/\Delta)$, where $R^2 > R$ is an approximate width of the potential stone location and $\Delta$ the stripe separation.

### 2.2. A generic continuous shape $\mathcal{M}$

MDL can be understood as an alternative regularization methodology for usual geometric fitting by minimizing the mean square error (MSE) of orthogonal distances $d(p,M)$, $p \in \mathcal{M}$. From various possible geometric regularization terms, the minimization of Gaussian curvature $\kappa_G$ leads to a very close relation to MDL minimization results. Equation 6 defines the $\kappa_G$ regularization:

$$l(M|\mathcal{M}) = \sum_{p \in \mathcal{M}} [d(p,M)]^2 + \lambda \sum_{p \in \mathcal{M}} \kappa_G p A_p \quad (6)$$

---

### Table 1. An informal summary of some geometric objects and their model costs. Uniform horizontal point cloud distribution has been assumed.

<table>
<thead>
<tr>
<th>$X$</th>
<th>$\Phi(X)$</th>
<th>$\phi_t(p)$</th>
<th>$n_1$</th>
<th>$n_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$O$</td>
<td>0</td>
<td>-</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$L_2$</td>
<td>2$\phi_3/3$</td>
<td>$\phi_3/3$</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>$L_3$</td>
<td>5$\phi_3/3$</td>
<td>$\phi_3/3$</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>$C$</td>
<td>2$\phi_3 + 1$</td>
<td>$\phi_3/3$</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>$P$</td>
<td>$\phi_3 + 1$</td>
<td>$2\phi_3/3$</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>$S$</td>
<td>4$\phi_3/3$</td>
<td>$2\phi_3/3 + 1$</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>$E$</td>
<td>3$\phi_3$</td>
<td>1...$2\phi_3/3 + 1$</td>
<td>9</td>
<td>26</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$X$</th>
<th>$\Phi(X)$</th>
<th>$\phi_t(p,X)$</th>
<th>$\phi_t(p,O)$</th>
<th>$n_1$</th>
<th>$n_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$O$</td>
<td>0</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>$L_2$</td>
<td>2$\phi_3/3$</td>
<td>$\phi_3/3$</td>
<td>$\phi_3/3$</td>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>$L_3$</td>
<td>5$\phi_3/3$</td>
<td>$\phi_3/3$</td>
<td>$\phi_3/3$</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>$C$</td>
<td>2$\phi_3 + 1$</td>
<td>$\phi_3/3$</td>
<td>$\phi_3/3$</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>$P$</td>
<td>$\phi_3 + 1$</td>
<td>$2\phi_3/3$</td>
<td>$\phi_3/3$</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>$S$</td>
<td>4$\phi_3/3$</td>
<td>$2\phi_3/3 + 1$</td>
<td>$\phi_3/3$</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>$E$</td>
<td>3$\phi_3$</td>
<td>1...$2\phi_3/3 + 1$</td>
<td>$\phi_3/3$</td>
<td>9</td>
<td>26</td>
</tr>
</tbody>
</table>

### Table 2. Minimum size of detectable stones.

| ground point density $\rho$ (1/m$^2$) | 0.7 | 42 |
| hemisphere min. radius | 2.3 | 0.3 |
| ellipsoid min. axis | 3.5 | 0.5 |
where the last sum represents an approximate integration of Gaussian curvature $\kappa_G$ over the surface of $M$ via discrete differential geometry (DDG) approach using a triangulation $T$ of the generic surface $M$ produced by the SAF algorithm of [1]. Approximation of the curvature $\kappa_G$ can be based on spatial angles of SAF using so called spherical excess (see [1]) or so called angle defect (see[5]). The curvature $\kappa_G$ is not available (or is not meaningful) for other objects in Table 1. This equals setting the regularization weight $\lambda \equiv 0$ for other target functions $l(X|\lambda')$ of the geometric fit. There are alternatives to the geometric regularization term, e.g. the number of triangles $|T|$ can be a cost term.

The MDL principle requires projection $\text{proj}(p, M)$ to the nearest surface triangle applied just as in Equations 2 and 3. The projectonal encoding length $\Phi_t(.)$ is based on the parameterization of each triangle $t \in T$ separately:

$$\Phi_t(\text{proj}(M, M)) = |M| |b(t)| + 3 |T| |b(|M|) | + (7)$$

$$+ 2 |M| |l(b(\sqrt{|M|}) / \epsilon)|,$$

where $|M| = \sum_{t \in T} \text{area}(t)$ is the total surface area of the object $M$. The term 1 of 7 refers to allocating of $|M|$ points to $|T|$ triangles. The term 2 is the encoding of the triangle information; each triangle consists of 3 points. The term 3 is about the encoding of each point by two local planar coordinates of some triangle $t \in T$.

For a practical implementation, one has to limit the shapes of triangles $t \in T$. The current attempt has triangles selected from a subset $M_T$ of points $M$ while so called Delaunay property [6] is being enforced. Details of this approach are still a topic of research.

### 3. TARGETING OF MDL TESTS

The site selection process is described in Figure 3. The first box generates a stoniness likelihood [1] map, which consists of $20 \times 20$ $m^2$ pixels. This likelihood map can be produced at areas where the ground hits exceed 60-70% of all ALS cloud points. This condition holds on most of the northern Finland. The first phase uses sparse ALS data and second phase dense UAV or ALS data (where available). The criterion for starting the MDL check at a specific point is based on the Gaussian curvature of the ground triangulation to have approximately constant value at neighboring ground points. Details are omitted in this presentation, and will be published later in an expansion paper. Red dots at Figure 2 represent prominent places to perform the MDL test. The quality of the local stone map shown in Figure 2 can be improved by neighborhood voting. Details of this process are still being developed. There are many possibilities and the goal is to develop a non-parametric adaptive process.

### 4. EXPERIMENTS AND RESULTS

A $220 \times 320 m$ test site with high-density point cloud has been used. The site is located at Harakkaaljalo at Turku, Finland. The coordinates are: $60.44^o \text{N}, 22.2^o \text{E}$.

![Figure 2. Point cloud pre-processing pipeline. Likely locations are scanned by a Gaussian curvature filter to pinpoint the MDL test spots. The scale of operations proceeds from 500 km to 20 m and 3 m.](image)

![Figure 3. Unfiltered Gaussian curvature distribution. Neighborhood voting would focus red stripes (possible stones) to red dots. The test site is in the city of Turku, Finland.](image)

A ground detail of this cloud is depicted in Figure 3. The cloud is exceptionally striped limiting the detection and radius estimation of the smallest stones. The stone B at Figure 3 has radius of $r = 0.4 m$, and it extends over one stripe only. The stone C is larger ($r = 1.2 m$), but the sample suffers from sparse ground hits because of the local thick canopy. There are some other small stones visible in the Figure 3.

The following treatise compares the root mean square error (RMSE) values $e$ to the average MDL encoding lengths $\phi = \Phi([G'])/|G'|$ with different point cloud densities. The upper density limit $\rho = 120 m^{-2}$ is typical for local photometric UAV scans. The site is covered by $\rho = 80 m^{-2}$ helicopter ALS scan, and the low limit is the nationwide open cloud with $\rho = 0.8 m^{-2}$. The low densities are simulated by uniform removal of points. In reality low density ALS scanners have larger beam radius and higher power thus having better penetration. This leads to somewhat different ground hit height distributions on each point cloud densities. The test side has s.d. $\sigma_{z} = 0.01...0.04 m$. The low range is for an urban grass field and the upper range is met at the thicket areas with high ground vegetation. Three stones were selected and both MSE and the average MDL encoding length $\phi$ calculated.

The top part of the Figure 4 shows that the assumption of a plane is simply a wrong one, when the analysis is tightly centered on a stone. But even the planarity assump-
5. CONCLUSION

This study is only rudimentary probing the possibilities of combining the existing stoniness likelihood pipeline of [1], preliminary narrowing of MDL test by neighborhood voting and the usability of the MDL test compared to MLE. Results, although preliminary, indicate that ground hit densities $\rho \geq 3 \, m^{-2}$ can be analyzed by the existing pipeline and MDL principle.

Another preliminary result is the size limit of the individually detectable stones in Table 2. To extend this treatise to have concrete scientific significance, the following tasks must be completed:

- Locating and measuring a set of stones at the test site for validation purposes.
- Choosing between various neighborhood voting methods to improve the likely stone locations seen in Figure 2.

Also, one has to experiment with the generic smooth shape $M$ either for detection of prominent ground objects or for generating an information theoretically justifiable alternative to the current triangulated ground model produced by SAF.

6. ACKNOWLEDGMENTS

Turku city arranged dense data sets from an accessible locality. Especially efforts of GIS engineer Asmo Leskinen are gratefully acknowledged.

7. REFERENCES

CHAINED KULLBACK-LEIBLER DIVERGENCES

Dmitri Pavlichin
Stanford University

ABSTRACT
We define and characterize the “chained” Kullback-Leibler divergence $\min_w D(p||w) + D(w||q)$ minimized over all intermediate distributions $w$ and the analogous $k$-fold chained K-L divergence $\min D(p||w_1) + D(w_1||w_2) + \ldots + D(w_k||q)$ minimized over the entire path $(w_1, \ldots, w_k)$. This quantity arises in a large deviations analysis of a Markov chain on the set of types – the Wright-Fisher model of neutral genetic drift: a population with allele distribution $q$ produces offspring with allele distribution $w$, which then produce offspring with allele distribution $p$, and so on.

The chained divergences enjoy some of the same properties as the K-L divergence (like joint convexity in the arguments) and appear in $k$-step versions of some of the same settings as the K-L divergence (like information projections and a conditional limit theorem). We further characterize the optimal $k$-step ”path” of distributions appearing in the definition and apply our findings in a large deviations analysis of the Wright-Fisher process. We make a connection to information geometry via the previously studied continuum limit, where the number of steps tends to infinity, and the limiting path is a geodesic in the Fisher information metric.

Finally, we offer a thermodynamic interpretation of the chained divergence (as the rate of operation of an appropriately defined Maxwell’s demon) and we state some natural extensions and applications (a $k$-step mutual information and $k$-step maximum likelihood inference). We release code for computing the objects we study.
TWO-FACED PROCESSES AND PSEUDORANDOM NUMBER GENERATORS

Boris Ryabko\textsuperscript{1} and Nadezhda Savina\textsuperscript{2}

\textsuperscript{1}Institute of Computational Technologies of Siberian Branch of RAS, Novosibirsk State University, Novosibirsk, Russia, boris@ryabko.net
\textsuperscript{2}Novosibirsk Humanitarian Institute, Novosibirsk, Russia, nnsavina@mail.ru

ABSTRACT

We describe binary-alphabet random processes whose Shannon entropy is less than 1 bit (per letter), but the frequency of occurrence of any word $u \in \{0, 1\}^*$ goes to $2^{-|u|}$. It gives a possibility to construct pseudorandom number generators which have proven properties. In turn, this possibility is important for applications such as those in cryptography.

We describe analogical processes for any finite alphabet and carried out some experiments in which low-entropy sequences are transformed into two-faced sequences.
BAYESIAN NETWORK STRUCTURE LEARNING WITH A QUOTIENT NORMALIZED MAXIMUM LIKELIHOOD CRITERION

Tomi Silander

Xerox Research Centre Europe
6 Chemin de Maupertuis
38242 Meylan, FRANCE
Tomi.Silander@xrce.xerox.fi

ABSTRACT

Learning the dependency structure of a multivariate distribution from the observational data is an important task since it allows us to speculate about the underlying causal mechanisms that induce dependencies. This learning task can be approached as a model selection problem, but recent studies have revealed that the popular Bayesian model selection criterion is not satisfactory. We review some of the information theoretic alternatives and introduce a new one called a quotient normalized maximum likelihood criterion.

1. INTRODUCTION

Bayesian networks [1] are popular models for presenting multivariate statistical dependencies that may have been induced by underlying causal mechanisms. Techniques for learning the structure of Bayesian networks from the observational data has therefore been used for many tasks such as discovering cell signaling pathways from the protein activity data [2], revealing the business process structures [3] from the transaction logs and modeling brain region connectivity using fMRI data [4].

Learning the structure of statistical dependencies can be seen as a model selection task where each model is a different hypothesis about the conditional dependencies between sets of variables. Traditional model selection criteria such as the Akaike information criterion [5] and the Bayesian information criterion [6] have also been used for the task, but recent comparisons have not been favorable for these criteria in terms of structural stability and/or predictive performance [7]. The most popular criterion has been the marginal likelihood (usually called BDeu for reasons explained later), but recent studies [8, 9] have found this criterion to be very sensitive to hyper parameters and to yield undesirably complex models for small sample sizes.

The information theoretic normalized maximum likelihood (NML) criterion [10, 11] would otherwise be an ideal candidate for a good criterion, but its exact calculation is likely to be prohibitively demanding. In 2008, Silander et al. introduced a hyper-parameter free, NML inspired criterion called a factorized NML (fNML) [7] that was shown to yield good predictive models without sensitivity problems. However, from the structure learning point of view, the fNML still sometimes appears to yield overly complex models. In this paper we introduce another NML related criterion, a quotient NML (qNML)

We will next briefly introduce Bayesian networks and then review the BDeu and fNML criteria and introduce the qNML criterion. We will also summarize the results for 20 datasets that back up our claim of qNML yielding parsimonious models with good predictive capabilities.

2. BAYESIAN NETWORKS

Bayesian networks are a general way to describe the dependencies between the components of an \( n \)-dimensional discrete data vector \( X = (X_1, \ldots, X_n) \) in which the component \( X_i \) may take any of the discrete values in a set \( \{1, \ldots, r_i\} \). Despite denoting the values with small integers, the model will treat the components of \( X \) categorically.

2.1. Likelihood

Bayesian network \( B = (G, \theta) \) defines a probability distribution for \( X \). The component \( G \) defines the structure of the model as a directed acyclic graph that has exactly one node for each component of \( X \). The structure \( G = (G_1, \ldots, G_n) \) defines for each variable/node \( X_i \) its (possibly empty) parent set \( G_i \), i.e., the nodes from which there are a directed edges to the variable \( X_i \).

Given a realization \( x \) of \( X \), we denote the sub-vector of \( x \) that consists of the values of the parents of \( X_i \) in \( x \) as \( G_i(x) \). It is customary to enumerate all the possible sub-vectors \( G_i(X) \) from 1 to \( q_i = \prod_{h \in G_i} r_h \). In case \( G_i \) is empty, we define \( q_i = 1 \) and \( P(G_i(x) = 1) = 1 \) for all vectors \( x \).

For each variable \( X_i \), there is a \( q_i \times r_i \) table \( \theta_i \) of parameters whose \( j^{th} \) row and \( k^{th} \) column defines the conditional probability \( P(X_i = k \mid G_j(X) = j; \theta) = \theta_{ijk} \). With structure \( G \) and parameters \( \theta \), we can now express the likelihood function of the model as

\[
P(x|G, \theta) = \prod_{i=1}^{n} P(x_i \mid G_i(x); \theta_i) = \prod_{i=1}^{n} \theta_{iG_i(x)x_i}, \quad (1)
\]

\[\text{We thank Teemu Roos for suggesting the name qNML.}\]
2.2. Bayesian structure learning

Bayesian learning of Bayesian network structures is based on the posterior probability \( P(G|D, \alpha) \), where \( \alpha \) denotes the hyper-parameters for the model parameters \( \theta \), and the \( D \) is a collection of \( N \times n \)-dimensional i.i.d. data vectors collected to a \( N \times n \) design matrix. We use structure \( G \) to extend the common matrix indexing notation \( D[i,j] \), and allow more general row selectors. Notably the notation \( D[G_i = j, i] \) is used to denote those rows of the column \( i \) in which the parents \( G_i \) contain the value configuration number \( j \in \{1, \ldots, q_i\} \).

It is common to assume the uniform prior for structures, in which case the objective function for structure learning is reduced to the marginal likelihood \( P(D|G, \alpha) \). If the model parameters \( \theta_{ij} \) are further assumed to be independently Dirichlet distributed only depending on \( i \) and \( G_i \), and the data \( D \) is assumed to have no missing values, the marginal likelihood can be decomposed as

\[
P(D|G, \alpha) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \int P(D[i,j]|\alpha) \, d\theta \approx \prod_{i=1}^{n} \prod_{j=1}^{q_i} P([G_i = j, i]|\theta) P(\theta|\alpha) d\theta.
\]

In coding terms this means that each data column \( D[i,\cdot] \) is first partitioned based on the values in columns \( G_i \), and each part is then coded using a Bayesian mixture that can be expressed in a closed form [12, 13].

2.3. Problems, solutions and problems

Finding the satisfactory Dirichlet hyperparameters for the Bayesian mixture above has, however, turned out to be problematic. Early on, one of the desiderata for a good model selection criterion was that it would yield equal scores for essentially equivalent models [14]. For example, the score for the structure \( X_1 \rightarrow X_2 \) should be the same as the score for the model \( X_2 \rightarrow X_1 \) since they both correspond to the hypothesis that variables \( X_1 \) and \( X_2 \) are statistically dependent on each other. It can be shown [13] that to achieve this, not all the hyperparameters \( \alpha \) are possible and for practical reasons Buntine [12] suggested a so called BDeu score with just one hyperparameter \( \alpha \in \mathbb{R}_{++} \) so that \( \theta_{ij} \sim Dir(\frac{1}{q_i}, \ldots, \frac{1}{q_i}) \). However, it soon turned out that BDeu score was very sensitive to the selection of this hyperparameter [8] and that for small sample sizes this method detects spurious correlations [9] leading to models with suspiciously many parameters.

A natural solution to avoid parameter sensitivity of BDeu would be to use a normalized maximum likelihood (NML) criterion [10, 11], i.e., to find the structure \( G \) that maximizes

\[
P_{\text{NML}}(D;G) = \frac{P(D|\hat{\theta}(D;G))}{\sum_{D'} P(D'|\hat{\theta}(D';G))}.
\]

where \( \hat{\theta} \) denotes the (easy to find) maximum likelihood parameters and the sum in the denominator goes over all the possible \( N \times n \) data matrices. While it is easy to see that this criterion satisfies the requirement of giving equal scores to equal structures, the normalizing constant renders the computation infeasible. Consequently, Silander et al. [7] suggested solving the parameter sensitivity problem by adopting the NML-code to the column partitions, i.e., changing the Bayesian mixture in equation (2) to

\[
P_{\text{NML}}^1(D[G_i = j, i];G) = \frac{P(D[\hat{\theta}(G_i = j, i);G])}{\sum_{D'} P(D';\hat{\theta}(D';G))},
\]

where \( D' \in \{1, \ldots, r_i\} \) is used to denote the rows of the column \( i \) in which the parents \( G_i \) contain the value configuration number \( j \in \{1, \ldots, q_i\} \). The logarithm of the denominator is often called a regret, since it indicates the extra code length needed compared to the code length obtained using the (a priori unknown) maximum likelihood parameters. The regret for \( P_{\text{NML}}^1 \) depends only on the length \( N \) of the categorical data vector with \( r \) different categorical values. While the naive computation of the regret is still prohibitive, it can be approximated efficiently using a so called Szpankowski approximation [15]:

\[
\text{reg}(N, r) = \sqrt{2r \Gamma \left( \frac{r}{2} \right)} \\
+ \frac{r - 1}{2} \log \left( \frac{N}{2} \right) - \log \left( \frac{r}{2} \right) + \frac{1}{2} \log (\pi) \\
+ \frac{2r^2 r^2 \left( \frac{r}{2} \right)}{36 N}. \tag{5}
\]

NML solves the parameter sensitivity problem and yields predictive models superior to BDeu. However, the criterion does not satisfy the property of giving the same score for the models that correspond to the same dependence statements. Furthermore, the learned structures still sometimes appear surprisingly complicated.

3. QUOTIENT NML SCORE

We will now introduce a quotient normalized maximum likelihood (qNML) criterion for learning Bayesian network structures. While equally efficient to compute than BDeu and fNML, it is free from hyperparameters, and it can be proven to give equal scores to equivalent models. Furthermore, it coincides with the actual NML score for exponentially many models. In our empirical tests it produces models featuring good predictive performance with significantly simpler structures than BDeu and fNML.

Like BDeu and fNML, qNML can be expressed as a product of \( n \) terms, one for each variable, but unlike the other two, it is not based on further partitioning the corresponding data column

\[
s_{\text{qNML}}^i(D;G) := \sum_{i=1}^{n} s_{\text{qNML}}^i(D;G) \tag{6}
\]

\[
:= \sum_{i=1}^{n} \frac{P_{\text{NML}}^1(D[i, i]; G)}{P_{\text{NML}}^1(D[i, i]; G)}
\]

The trick here is to model a subset of columns as though there were no conditional independencies among the corresponding variables \( S \subset X \). In this case, we can collapse
the $\prod_{X \in S} r_i$ value configurations and consider them as values of a single variable with $\prod_{X \in S} r_i$ different values which can then be modeled with a one-dimensional $P^1_{\text{NML}}$ code. The $sq_{\text{NML}}$ does not necessarily define a distribution for $D$, but it is easy to verify that it coincides with the $P_{\text{NML}}(D; G)$ for all the networks that are composed of fully connected components. The number of such networks equals the number of nonempty partitions of a set of $n$ elements, i.e., the $n^{\text{th}}$ Bell number.

4. EXPERIMENTAL RESULTS

To empirically compare the model selection criteria we took 20 UCI data sets [16] and ran 1000 train and test experiments for 20 different sample sizes for all of them. The training was conducted using a dynamic programming based exact structure learning [17] which limited the number $n$ of variables to less than 20.

Figure 1 shows an example of the behavior of the studied selection criteria. For small sample sizes fNML and qNML yield equally accurate predictions while BDeu requires more samples to obtain competitive performance. The curves are mean curves from 1000 independent train and test splits.

Figure 2 shows how fNML still sometimes behaves strangely in terms of model complexity here measured by the number of parameters in the model. qNML, instead, appears to yield more parsimonious models.

Table 1 features predictive losses for the small sample sizes. (The results for large sample sizes converge for all sensible criteria.) As we can see, fNML still usually (12/20) performs best but qNML is not much worse while BDeu may sometimes perform much worse.

Table 2. Number of model parameters for small sample sizes for different model selection criteria in 20 different datasets.

<table>
<thead>
<tr>
<th>Data</th>
<th>N</th>
<th>BDeu</th>
<th>fNML</th>
<th>qNML</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iris</td>
<td>16</td>
<td>3.59</td>
<td>3.47</td>
<td>3.43</td>
</tr>
<tr>
<td>PostOperative</td>
<td>20</td>
<td>10.24</td>
<td>8.27</td>
<td>8.32</td>
</tr>
<tr>
<td>Wine</td>
<td>27</td>
<td>11.81</td>
<td>12.68</td>
<td>12.51</td>
</tr>
<tr>
<td>HeartHungarian</td>
<td>45</td>
<td>10.15</td>
<td>9.10</td>
<td>8.90</td>
</tr>
<tr>
<td>BreastCancer</td>
<td>45</td>
<td>12.01</td>
<td>10.57</td>
<td>10.66</td>
</tr>
<tr>
<td>HeartCleveland</td>
<td>48</td>
<td>16.36</td>
<td>12.81</td>
<td>12.54</td>
</tr>
<tr>
<td>Ecoli</td>
<td>51</td>
<td>5.77</td>
<td>5.30</td>
<td>5.37</td>
</tr>
<tr>
<td>Liver</td>
<td>54</td>
<td>4.07</td>
<td>3.97</td>
<td>3.91</td>
</tr>
<tr>
<td>Glass</td>
<td>55</td>
<td>7.16</td>
<td>6.22</td>
<td>6.28</td>
</tr>
<tr>
<td>Thyroid</td>
<td>55</td>
<td>2.83</td>
<td>2.75</td>
<td>2.77</td>
</tr>
<tr>
<td>HeartStatlog</td>
<td>56</td>
<td>14.15</td>
<td>11.92</td>
<td>11.60</td>
</tr>
<tr>
<td>TicTacToe</td>
<td>96</td>
<td>10.99</td>
<td>10.73</td>
<td>11.05</td>
</tr>
<tr>
<td>Balance</td>
<td>96</td>
<td>7.48</td>
<td>7.39</td>
<td>7.78</td>
</tr>
<tr>
<td>BCWisconsin</td>
<td>105</td>
<td>5.81</td>
<td>5.12</td>
<td>5.08</td>
</tr>
<tr>
<td>Diabetes</td>
<td>117</td>
<td>4.95</td>
<td>4.95</td>
<td>4.89</td>
</tr>
<tr>
<td>Yeast</td>
<td>225</td>
<td>5.54</td>
<td>5.40</td>
<td>5.42</td>
</tr>
<tr>
<td>Abalone</td>
<td>418</td>
<td>3.93</td>
<td>3.88</td>
<td>3.96</td>
</tr>
<tr>
<td>PageBlocks</td>
<td>548</td>
<td>2.36</td>
<td>2.32</td>
<td>2.38</td>
</tr>
<tr>
<td>Shuttle</td>
<td>2900</td>
<td>1.70</td>
<td>1.69</td>
<td>1.72</td>
</tr>
<tr>
<td>Adult</td>
<td>4885</td>
<td>10.22</td>
<td>10.08</td>
<td>10.12</td>
</tr>
</tbody>
</table>

Table 1. Predictive log losses for small sample sizes for different model selection criteria in 20 different datasets.

<table>
<thead>
<tr>
<th>Data</th>
<th>N</th>
<th>BDeu</th>
<th>fNML</th>
<th>qNML</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iris</td>
<td>16</td>
<td>36</td>
<td>33</td>
<td>28</td>
</tr>
<tr>
<td>PostOperative</td>
<td>20</td>
<td>1304</td>
<td>464</td>
<td>115</td>
</tr>
<tr>
<td>Wine</td>
<td>27</td>
<td>11599</td>
<td>780</td>
<td>214</td>
</tr>
<tr>
<td>HeartHungarian</td>
<td>45</td>
<td>4706</td>
<td>838</td>
<td>100</td>
</tr>
<tr>
<td>BreastCancer</td>
<td>45</td>
<td>26774</td>
<td>2710</td>
<td>246</td>
</tr>
<tr>
<td>HeartCleveland</td>
<td>48</td>
<td>29346</td>
<td>1407</td>
<td>204</td>
</tr>
<tr>
<td>Ecoli</td>
<td>51</td>
<td>113</td>
<td>167</td>
<td>80</td>
</tr>
<tr>
<td>Liver</td>
<td>54</td>
<td>40</td>
<td>60</td>
<td>24</td>
</tr>
<tr>
<td>Glass</td>
<td>55</td>
<td>1488</td>
<td>572</td>
<td>100</td>
</tr>
<tr>
<td>Thyroid</td>
<td>55</td>
<td>37</td>
<td>70</td>
<td>28</td>
</tr>
<tr>
<td>HeartStatlog</td>
<td>56</td>
<td>18414</td>
<td>2438</td>
<td>696</td>
</tr>
<tr>
<td>TicTacToe</td>
<td>96</td>
<td>13645</td>
<td>237</td>
<td>168</td>
</tr>
<tr>
<td>Balance</td>
<td>96</td>
<td>20</td>
<td>237</td>
<td>168</td>
</tr>
<tr>
<td>BCWisconsin</td>
<td>105</td>
<td>5921</td>
<td>816</td>
<td>88</td>
</tr>
<tr>
<td>Diabetes</td>
<td>117</td>
<td>37</td>
<td>216</td>
<td>34</td>
</tr>
<tr>
<td>Yeast</td>
<td>225</td>
<td>78</td>
<td>307</td>
<td>80</td>
</tr>
<tr>
<td>Abalone</td>
<td>418</td>
<td>90</td>
<td>156</td>
<td>62</td>
</tr>
<tr>
<td>PageBlocks</td>
<td>548</td>
<td>699</td>
<td>777</td>
<td>56</td>
</tr>
<tr>
<td>Shuttle</td>
<td>2900</td>
<td>380</td>
<td>594</td>
<td>111</td>
</tr>
<tr>
<td>Adult</td>
<td>4885</td>
<td>599</td>
<td>1431</td>
<td>876</td>
</tr>
</tbody>
</table>

Table 2. Number of model parameters for small sample sizes for different model selection criteria in 20 different datasets.

![Figure 1](image1.png)

Figure 1. Predictive log-loss in a breast cancer data as a function of sample size for different model selection criteria.

![Figure 2](image2.png)

Figure 2. Number of parameters in a breast cancer model as a function of sample size for different model selection criteria.
Looking at the number of parameters for the same datasets and sample sizes (Table 2) reveals that qNML usually (18/20) yields simplest models, and sometimes, like in the BreastCancer data, the differences are of orders of magnitude.

5. CONCLUSION
We have presented qNML, a new model selection criterion for learning structures of Bayesian networks. While being competitive in predictive terms, it often yields significantly simpler models than other common model selection criteria. The computational cost of qNML equals the cost of earlier criteria. The criterion also gives equal scores for models that encode same independence hypotheses about the joint probability distribution. qNML also coincides with NML criterion for exponentially many models.

6. REFERENCES
JEFFREYS’ AND BDEU PRIORS FOR MODEL SELECTION

Joe Suzuki
Department of Mathematics, Osaka University, suzuki@math.sci.osaka-u.ac.jp

ABSTRACT
The goal of this paper is to compare BDeu (Bayesian Dirichlet-
let equivalent uniform) and Jeffreys’ priors for model se-
lection. The main contribution is to prove that given three
sequences \(x^n = (x_1, \cdots, x_n)\), \(y^n = (y_1, \cdots, y_n)\), \(z^n = (z_1, \cdots, z_n)\) of length \(n\), BDeu rejects \(X \perp Y | Z\) if
\(H(x^n|z^n) = 0\) and \(H(y^n|z^n) = 0\) while Jeffreys accepts
\(X \perp Y | Z\) if \(H(x^n|z^n) = 0\) or \(H(y^n|z^n) = 0\), where
\(H(\cdot|\cdot)\) denotes the conditional empirical entropy.

Extended Abstract
Suppose that given \(x^n = (x_1, \cdots, x_n) \in \{0, 1\}^n\), we
assign a probability \(Q(x^n)\) to each binary sequence \(x^n\)
of length \(n\). It is known that if we prepare constants \(a, b > 0\)
to compute the quantity
\[
Q(x^n) = \int_0^1 P(x^n|\theta)w(\theta)d\theta
\]
for the \(2^n\) sequences, where \(P(x^n|\theta) = \theta^c(1-\theta)^n-c\) (c: oc-
currence of ones out of \(n\)) and \(w(\theta) \propto \theta^{a-1}(1-\theta)^{b-1}\),
the quantity \(-\log Q(x^n)\) converges to its entropy as \(n\) grows with probability one. We consider to construct
quantity \(Q(\cdot)\) to estimate conditional independence state-
ments among variables.

Let \(a(x_i) := \begin{cases} a & (x_i = 0) \\ b & (x_i = 1) \end{cases}\) and \(c_{i-1}(x^i)\) the oc-
currence of \(x_i = 0, 1\) in \((x_1, \cdots, x_{i-1})\). Then, we have the
following expression: \(Q(x^n) = \prod_{i=1}^n c_{i-1}(x_i) + a(x_i)\).

For example, if \(a = 0.1, b = 0.2,\) and \((x_1, \cdots, x_n) =
(0, 1, 0, 1, 1),\) then
\[
Q(x^n) = \frac{0 + 0.1}{0 + 0.3} \cdot \frac{0 + 0.2}{1 + 0.3} \cdot \frac{1 + 0.1}{2 + 0.3} \cdot \frac{1 + 0.2}{3 + 0.3} \cdot \frac{2 + 0.2}{4 + 0.3}
\]
The idea can be extended from binary \(\{0, 1\}\) to \(\alpha\)-nary
\(\{0, 1, \cdots, \alpha - 1\}\) \((\alpha \geq 2)\).

Next, we consider to assign a probability to each pair
of \((X, Y)\) rather than to a single \(X\): given sequences \(x^n =
(x_1, \cdots, x_n) \in \{0, 1, \cdots, \alpha - 1\}^n \) \((\alpha \geq 2)\) and \(y^n =
(y_1, \cdots, y_n) \in \{0, 1, \cdots, \beta - 1\}^n \) \((\beta \geq 2)\), we construct
a similar quantity:
\[
Q(x^n, y^n) = \int P(x^n, y^n|\theta)w(\theta)d\theta
\]
where \(a(x, y) > 0\) and \(c_{i-1}(x, y)\) are a constant associ-
ated with \(X = x, Y = y\) and the occurrence of \(X = x\) and
\(Y = y\) in \(x^{i-1}\) and \(y^{i-1}\), respectively, and \(P(x^n, y^n|\theta)\)
and \(w(\theta)\) are given by
\[
P(x^n, y^n|\theta) = \prod_{x,y} \theta(x,y)^c(x,y)
\]
and
\[
w(\theta) \propto \prod_{x,y} \theta(x,y)^{a(x,y) - 1},
\]
respectively. For example, if \(a(x,y)\) and \((x^n, y^n)\) are
given by the following table,
\[
\begin{array}{cccccc}
  x & y & 0 & 1 & 2 & 3 \\
  0 & 0.1 & 0.2 & 0.3 & 0.4 & 0.5 \\
  1 & 0.1 & 0.2 & 0.3 & 0.4 & 0.5 \\
\end{array}
\]
we can compute \(Q(x^n, y^n)\) by
\[
0 + 0.2, 0 + 0.4, 0 + 0.1, 0 + 0.3, 1 + 0.4
\]
\[
0 + 0.2, 0 + 0.4, 0 + 0.1, 0 + 0.3, 1 + 0.4
\]

Furthermore, we can define its conditional alternative
as follows:
\[
Q(y^n|x^n) = \int P(y^n|x^n, \theta)w(\theta)d\theta = \prod_x Q(y^n|x)
\]
\[
= \prod_x \left( \prod_{i=x} c_{i-1}(x_i, y_i) + a(x_i, y_i) \right).
\]
For example, if \(a(x, y)\) and \((x^n, y^n)\) are the same as be-
fore, we can compute \(Q(y^n|x)\) and \(Q(y^n|x)\)
and \(Q(y^n|x)\) as before, and
\[
Q(y^n|x) = \begin{cases} 0 + 0.2 & 0 + 0.1 \\
0 + 0.3 & 1 + 0.3 \end{cases}
\]
\[
0 + 0.4 & 0 + 0.3 \\
0 + 0.7 & 1 + 0.7 \\
2 + 0.7 & 2 + 0.7 \\
\]
\[
0 + 0.3 & 0 + 0.1 \\
0 + 0.7 & 1 + 0.7 \\
2 + 0.7 & 2 + 0.7 \\
\]
\[
\]
In this paper, we define two choices of \(a(x, y)\) as fol-
"
Table 1. BDeu and Jeffreys Priors (Conditional Probabilities)

<table>
<thead>
<tr>
<th></th>
<th>(Q(x^n))</th>
<th>(Q(y^n))</th>
</tr>
</thead>
<tbody>
<tr>
<td>BDeu</td>
<td>(\frac{\prod_{i=1}^{\delta} c_{i-1}(x_i) + \delta/\alpha}{\alpha/\beta} )</td>
<td>(\prod_{x} \left{ \prod_{i:x=x_i} c_{i-1}(x_i, y_i) + \delta/\alpha \right} )</td>
</tr>
<tr>
<td>Jeffreys</td>
<td>(\frac{\prod_{i=1}^{0.5} c_{i-1}(x_i) + 0.5}{i - 1 + 0.5\alpha} )</td>
<td>(\prod_{x} \left{ \prod_{i:x=x_i} c_{i-1}(x_i, y_i) + 0.5 \right} )</td>
</tr>
</tbody>
</table>

BDeu and Jeffreys, respectively. More precisely, we summarize them in Table 1. Then, one can check that BDeu satisfies score equivalence:

\[
Q(x^n)Q(y^n|x^n) = Q(x^n, y^n) = Q(y^n)Q(x^n|y^n)
\]

(n general, \(Q(x^n)Q(y^n|x^n) \neq Q(y^n)Q(x^n|y^n)\).

In this paper, without using

\[
Q(y^n|x^n) = \prod_{x} \left\{ \prod_{i:x=x_i} c_{i-1}(x_i, y_i) + a(x, y) \right\},
\]

we redefine

\[
Q(y^n|x^n) := \frac{Q(x^n, y^n)}{Q(x^n)}
\]

so that score equivalence is guaranteed for Jeffreys' prior as well. In particular, we use simultaneous probabilities and their quotients rather than the conditional probabilities (Table 2).

As an application, if we compare and choose the structure that maximizes

\[
Q(x^n)Q(y^n)Q(z^n) , \quad Q(x^n)Q(y^n, z^n),
\]

\[
Q(y^n)Q(z^n, x^n) , \quad Q(z^n)Q(x^n, y^n),
\]

\[
\frac{Q(x^n, y^n)Q(y^n, z^n)}{Q(y^n)} , \quad \frac{Q(z^n, x^n)Q(x^n, y^n)}{Q(z^n)}
\]

\[
\frac{Q(y^n)Q(z^n)Q(x^n, y^n, z^n)}{Q(y^n, z^n)} , \quad \frac{Q(z^n)Q(x^n)Q(x^n, y^n, z^n)}{Q(z^n, x^n)}
\]

\[
\frac{Q(x^n)Q(y^n)Q(z^n, x^n, y^n, z^n)}{Q(x^n, y^n)} , \quad Q(x^n, y^n, z^n)
\]

multiplied by the prior probabilities, then we obtain the Bayesian network (one of the eleven in Figure 1) with the maximum posterior probability.

Another example is to estimate \(X \perp \perp Y\) given by \(x^n\) and \(y^n\). Let \(p\) be the prior probability of \(X \perp \perp Y\), and \(I(X, Y)\) the mutual information between \(X\) and \(Y\) (\(I(X, Y) = 0 \iff X \perp \perp Y\)). If we follow the rule

\[
pQ(x^n)Q(y^n) \leq (1-p)Q(x^n, y^n) \iff X \perp \perp Y,
\]

then the estimation is correct with probability one as \(n \to \infty\). Noting the fact, Suzuki 1993,2012 proposed a mutual information estimator

\[
J(n) := \frac{1}{n} \log \left( \frac{(1-p)Q(x^n, y^n)}{pQ(x^n)Q(y^n)} \right)
\]

\[
\approx I(n) - \frac{(\alpha - 1)(\beta - 1)}{2} \log n \to I(X, Y),
\]

where \(I(n)\) is the plug-in mutual information estimator.

Now we consider the following example: \(n = 5, \delta = 1, x^5 = (0, 0, 0, 0, 0), y^5 = (0, 0, 0, 0, 0), p = 0.5\). For BDeu and Jeffreys

\[
\frac{Q(x^n) = Q(y^n)}{0 + 0.5} = \frac{1 + 0.5}{1 + 1} \cdot \frac{2 + 0.5}{2 + 2} \cdot \frac{3 + 0.5}{3 + 1} \cdot \frac{4 + 0.5}{4 + 4} = 0.246
\]

\[
Q(x^n)Q(y^n) = 0.246^2 = 0.0605 .
\]

However, \(J(n) > 0\) for BDeu (\(\delta = 1\)):

\[
\frac{Q(x^n, y^n)}{\prod_{i=1}^{n} c_{i-1}(x_i, y_i) + \delta/\alpha \beta} \frac{\prod_{i=1}^{0.5} c_{i-1}(x_i) + 0.5}{i - 1 + 0.5\alpha} \]

\[
= \frac{0 + 1/4}{1 + 1/4} \cdot \frac{2 + 1/4}{2 + 2} \cdot \frac{3 + 1/4}{3 + 1} \cdot \frac{4 + 1/4}{4 + 4} \]

\[
= 0.0809 \cdots
\]

while \(J(n) \leq 0\) for Jeffreys:

\[
\frac{Q(x^n)Q(y^n)}{\prod_{i=1}^{n} c_{i-1}(x_i, y_i) + 0.5} \frac{\prod_{i=1}^{0.5} c_{i-1}(x_i) + 0.5}{i - 1 + 0.5\alpha \beta} \]

\[
= \frac{0 + 0.5}{1 + 0.5} \cdot \frac{2 + 0.5}{2 + 2} \cdot \frac{3 + 0.5}{3 + 1} \cdot \frac{4 + 0.5}{4 + 4} \]

\[
= 0.0410 \cdots
\]

Moreover, we find that for each \(n > 1\)

\[
x_1 = \cdots = x_n \text{ and } y_1 = \cdots = y_n \implies J(n) > 0 \text{ for BDeu}
\]

\[
x_1 = \cdots = x_n \text{ or } y_1 = \cdots = y_n \implies J(n) \leq 0 \text{ for Jeffreys }.
\]

In general, let \(x^n = (x_1, \cdots, x_n), y^n = (y_1, \cdots, y_n), \)

\(z^n = (z_1, \cdots, z_n), \) and

\[
J(n) := \log \frac{Q(x^n, y^n, z^n)}{Q(x^n, z^n) \cdot Q(y^n, z^n)}
\]

The result for independence is extended to that for conditional independence.

**Theorem 1** For each \(n \geq 1\), if \(H(x^n|z^n) = 0\) and \(H(y^n|z^n) = 0\), then \(J(n) > 0\) for BDeu.

**Theorem 2** For each \(n \geq 1\), if \(H(x^n|z^n) = 0\) or \(H(y^n|z^n) = 0\), then \(J(n) \leq 0\) for Jeffreys.
Table 2. BDeu and (modified) Jeffreys Priors (Simultaneous Probabilities)

|        | \(a(x, y)\) | \(Q(x^n)\) | \(Q(y^n|x^n)\) |
|--------|-------------|-------------|---------------|
| BDeu   | \(\frac{\delta}{\alpha\beta}\) | \(\prod_{i=1}^{n} \frac{c_{i-1}(x_i) + \delta/\alpha}{i - 1 + \delta}\) | \(\prod_{x} \prod_{i:x=x_i} \frac{c_{i-1}(x_i, y_i) + \delta/\alpha\beta}{c_{i-1}(x_i) + \delta/\alpha}\) |
| Jeffreys| 0.5 | \(\prod_{i=1}^{n} c_{i-1}(x_i) + 0.5\) | \(\prod_{x} \prod_{i:x=x_i} c_{i-1}(x_i, y_i) + 0.5\) |

Note that \(H(x^n|z^n) = 0\) if and only if there exists a function \(f\) s.t. \(x_i = f(z_i)\) for all \(i = 1, \ldots, n\).

For example, the following dataset satisfies the conditions of Theorems 1 and 2. In fact, we observe \(z_i = 0, 1, \Rightarrow x_i = 0, z_i = 2, 3, \Rightarrow x_i = 1, z_i = 0, 2, \Rightarrow y_i = 0,\) and \(z_i = 1, 3, \Rightarrow y_i = 1,\) thus \(H(x^n|z^n) = H(y^n|z^n) = 0.\) From Theorems 1 and 2, we have \(J(n) > 0\) and \(J(n) \leq 0\) for BDeu and Jeffreys, which means each of them rejects and accepts \(X \perp \perp Y|Z,\) respectively.

\[
\begin{array}{c}
\begin{array}{ccc}
Y & X & Z \\
\hline
1 & 2 & 3 \\
0 & 0 & 1 \\
0 & 1 & 0 \\
0 & 1 & 2 \\
\end{array}
& \begin{array}{ccc}
Y & Z & X \\
\hline
1 & 2 & 3 \\
0 & 0 & 1 \\
0 & 1 & 0 \\
0 & 1 & 2 \\
\end{array}
& \begin{array}{ccc}
Y & Z & \emptyset \\
\hline
1 & 2 & 3 \\
0 & 0 & 1 \\
0 & 1 & 0 \\
0 & 1 & 2 \\
\end{array}
\end{array}
\]

Then, from Theorems 1 and 2, BDeu and Jeffreys estimate

\[
\frac{Q(x^n, y^n, z^n)}{Q(y^n, z^n)} > \frac{Q(x^n, z^n)}{Q(z^n)}
\]

and

\[
\frac{Q(x^n, y^n, z^n)}{Q(y^n, z^n)} \leq \frac{Q(x^n, z^n)}{Q(z^n)}
\]

and choose \(\{Y, Z\}\) and \(\{Y\}\), as the parent sets of \(X,\) respectively (Figure 2). However, if we follow regularity and notice that both of \(\{Y, Z\}\) and \(\{Y\}\) optimize the fitness (in the sense of the conditional empirical entropy being zero), we choose \(\{Y\}\) because the simplicity is better. In this sense, BDeu violates regularity while Jeffreys follows.

Some might think that a prior probability is based on his or her belief, and that nobody should reject it from a general point of view. However, the fact that was proved in this paper has not been known thus far, and people that have been holding BDeu as their priors may change them to beliefs that follows regularity.

Appendix A: Proofs

\(J(n) > 0\) for BDeu

For \(\delta > 0,\) since

\[
Q(x^n)Q(y^n) = \frac{\Gamma(\delta)}{\Gamma(n + \delta)} \prod_{x} \frac{\Gamma(c(x) + \delta/\alpha)}{\Gamma(\delta/\alpha)}
\]

and

\[
Q(x^n, y^n) = \frac{\Gamma(\delta)}{\Gamma(n + \delta)} \prod_{x} \prod_{y} \frac{\Gamma(c(x, y) + \delta/\alpha\beta)}{\Gamma(\delta/\alpha\beta)}
\]

Figure 1. Eleven Markov equivalent Bayesian networks with three nodes
if we assume $x_1 = \cdots = x_n$ and $y_1 = \cdots = y_n$, it is sufficient to show
\[ \frac{\Gamma(n + \delta/\alpha)}{\Gamma(\delta/\alpha)} \cdot \frac{\Gamma(n + \delta/\alpha)}{\Gamma(\delta/\alpha)} \leq \frac{\Gamma(n + \delta/\alpha \beta)}{\Gamma(\delta/\alpha \beta)} \frac{\Gamma(n + \delta)}{\Gamma(\delta)} . \]

This can be proved using induction for $n \geq 1$.

\[ J(n) \leq 0 \text{ for Jeffreys} \]

Since
\[ Q(x^n)Q(y^n) = \frac{\Gamma(\alpha/2)}{\Gamma(n + \alpha/2)} \prod_x \frac{\Gamma(c(x) + 1/2)}{\Gamma(1/2)} \cdot \frac{\Gamma(\beta/2)}{\Gamma(n + \beta/2)} \prod_y \frac{\Gamma(c(y) + 1/2)}{\Gamma(1/2)} , \]

and
\[ Q(x^n, y^n) = \frac{\Gamma(\alpha/2 \beta/2)}{\Gamma(n + \alpha/2 \beta/2)} \prod_x \prod_y \frac{\Gamma(c(x, y) + 1/2)}{\Gamma(1/2)} , \]

if we assume $x_1 = \cdots = x_n$, it is sufficient to show
\[ \frac{\Gamma(\alpha/2)}{\Gamma(n + 1/2)} \frac{\Gamma(n + 1/2)}{\Gamma(1/2)} \cdot \frac{\Gamma(\beta/2)}{\Gamma(n + \beta/2)} \prod_y \frac{\Gamma(c(y) + 1/2)}{\Gamma(1/2)} \geq \frac{\Gamma(\alpha/2 \beta/2)}{\Gamma(n + \alpha/2 \beta/2)} \prod_y \frac{\Gamma(c(y) + 1/2)}{\Gamma(1/2)} , \]

which means
\[ \frac{\Gamma(n + \alpha/2 \beta/2)}{\Gamma(n + 1/2)} \frac{\Gamma(n + 1/2)}{\Gamma(1/2)} \geq \frac{\Gamma(n + \alpha/2)}{\Gamma(\alpha/2)} \frac{\Gamma(n + \beta/2)}{\Gamma(\beta/2)} . \]

This can be proved using induction for $n \geq 0$.

**An Intuitive Reasoning of the result**

For Jeffreys’, there exist constants $C_1, C_2$ (uniformly bounded) s.t.
\[ I(n) - \frac{(\alpha - 1)(\beta - 1)}{n} \log n + \frac{C_1}{n} \leq J(n) \leq I(n) - \frac{(\alpha - 1)(\beta - 1)}{n} \log n + \frac{C_2}{n} . \]

On the other hand, for BDeu, there exist $D_n \rightarrow D(\theta)$ (not uniformly) s.t.
\[ J(n) \leq I(n) - \frac{(\alpha - 1)(\beta - 1)}{n} \log n + \frac{D_n}{n} . \]

For example, For BDeu with $\alpha = \beta = 2$ and $\delta = 1$,
\[ P(X = 1) = P(Y = 1) = n^{-0.75} \text{ and } X \perp \!\!\!\!\perp Y \implies D_n > 0.5 \times \log n \implies I(n) \approx 0 \text{ and } J(n) \approx -\frac{1}{2n} \log n + \frac{D_n}{n} > 0 \]

For each $n$, there exists $\theta$ s.t. $X \perp \!\!\!\!\perp Y$ and $J(n) > 0$ (See Figure 3).
SPARSE MODELLING USING INFORMATION THEORETIC STRUCTURE SELECTION

Ioan Tabus
Tampere University of Technology

ABSTRACT
We exemplify the use of MDL model structure selection in sparse models, for the case where the signal or image exhibit quasi-periodic regions or closely approximate-matching regions. We show that designing sparse predictors using greedy algorithms is providing good interpolation models, without the need of using laborious matching search between regions.
RATE-DISTORTION THEORETIC VIEWS OF LEARNING PROBLEMS

Kazuho Watanabe
Department of Computer Science and Engineering, Toyohashi University of Technology, 1-1 Hibrigaoka Tempaku-cho Toyohashi, Aichi 441-8580, JAPAN, wkazuho@cs.tut.ac.jp

ABSTRACT
Rate-distortion theory analyzes fundamental limits of lossy compression methods. In this study, we discuss the connections of the rate-distortion theory to Bayesian learning and a clustering method obtained from Bayesian nonparametrics.

1. INTRODUCTION
The rate-distortion (RD) theory studies and characterizes the performance of lossy compression systems by the RD function [1]. There are studies interpreting learning and prediction problems by rate-distortion theoretic views. In particular, clustering methods can be considered as vector quantizers and naturally related to the RD theory [2]. RD theoretic interpretations have been obtained also for problems such as classification [3] and sequence prediction [4].

In this study, we provide a rate-distortion theoretic view of Bayesian learning, and show that its optimal solution is given by the generalized posterior distribution, which has been used for purposes such as the posterior consistency [5, 6] and model selection [7, 8, 9]. We also discuss the connection of this view to the asymptotic theory of Bayesian learning, which characterizes the generalization error of a learning machine by a constant called learning coefficient [10]. It is demonstrated that the learning coefficient provides an upper bound of the RD dimension [11] of this RD problem. We also provide a RD theoretic interpretation of the Dirichlet process (DP) means clustering, which is a simple extension of the K-means clustering and estimates the number of clusters from data [12, 13].

2. RATE-DISTORTION FUNCTION
Let X and Y be random variables of input and reconstruction taking values in X and Y, respectively. For the nonnegative distortion measure between x and y, d(x, y), the rate-distortion function R(D) of the source X ∼ p(x) is defined by

\[ R(D) = \inf_{q(y|x):E[d(X,Y)] \leq D} I(q(y|x)), \]

where

\[ I(q(y|x)) = \int \int q(y|x)p(x) \log \frac{q(y|x)}{q(y|x)p(x)} dx dy \]
is the mutual information between X and Y and E denotes the expectation with respect to q(y|x)p(x). R(D) shows the minimum achievable rate for the i.i.d. source with the density p(x) under the given distortion measure d [1, 14].

If there exists the conditional distribution \( q^*_s \) that achieves the minimum of the following free energy functional parameterized by s ≥ 0,

\[ F(q(y)) = E_p \left[ -\log \int \exp(-sd(X,y))q(y)dy \right], \]

then, the rate-distortion function is parametrically given by

\[ R(D_s) = I(q^*_s(y|x)), \]

\[ D_s = E_{q^*_s(y|x)p(x)}[d(X,Y)], \]

where

\[ q^*_s(y|x) = \frac{\exp(-sd(x,y))q^*_s(y)}{\int \exp(-sd(x,y))q^*_s(y)dy}. \]

The parameter s corresponds to the (negated) slope of the tangent of R(D) at (D_s, R(D_s)) and hence is referred to as the slope parameter [1]. From the properties of the rate-distortion function R(D), we know that R(D) > 0 for 0 < D < D_{max}, where

\[ D_{max} = \inf_y E_p[d(X,y)], \]

and R(D) = 0 for D ≥ D_{max} [1].

3. BAYESIAN LEARNING
Given training data x^n = \{x_1, \ldots, x_n; x_i \in X_0, \forall i\}, we assume the statistical model \{p(x|\theta); \theta \in \Theta \subset \mathbb{R}^m\} with an m-dimensional parameter \theta. We further assume that the data are i.i.d., \( p(x^n) = \prod_{i=1}^n p(x_i) \) and \( p(x^n|\theta) = \prod_{i=1}^n p(x_i|\theta) \) and that the maximum likelihood estimator (MLE), \( \hat{\theta}(x^n) \), for the data x^n exists for all x^n.

3.1. RD-Theoretic Framework
Let us consider learning \( \theta \) from x^n through the following RD problem. The training data set x^n ∈ X^n_0 is an instance of the source output, \( X = X^n_0 \), and the parameter \( \theta \) is viewed as the reconstruction of x^n, \( Y = \Theta \). The distortion measure between them is defined by

\[ d(x^n, \theta) = \log \frac{p(x^n|\hat{\theta}(x^n))}{p(x^n|\theta)}, \]
the regret of θ for the data set xn.
Assume that there exists
θ0 = argmin
θ
KL(p(x)||p(x|θ)),
the minimizer of the Kullback-Leibler divergence from the source. Then, the minimum distortion at zero rate is
\[ D_{\max} = E_p[\log p(X^n|\hat{\theta}(X^n))] - E_p[\log p(X^n|\theta_0)], \] (3)
where the second term reduces to the source entropy, nh(p) = nE_p[-log p(X)] if the source is realizable by the model, p(x) = p(x|θ0). D_{\max} in (3) is the (negative) training error of the MLE normalized by the second term.

For the distortion levels smaller than D_{\max}, if there exists
\[ q^*_n(\theta) = \arg\min_{q(\theta)} E_p \left[ -\log \int p(X^n|\theta)^s q(\theta)d\theta \right], \]
then the RD function is given by
\[ R(D_s) = E_p \left[ -\log \int p(X^n|\theta)^s q^*_n(\theta)d\theta \right] + sE_p[\log p(X^n|\theta)], \]
\[ D_s = E_p[\log p(X^n|\hat{\theta}(X^n)) - \log p(X^n|\theta)], \]
where
\[ q^*_n(\theta|x^n) \propto p(x^n|\theta)^s q^*_n(\theta). \]

### 3.2. Learning Coefficient and RD Dimension

If we fix a prior distribution q(θ), as in the usual Bayesian learning, then we have the following upper bound of R(D),
\[ R(D_s) \leq E_p \left[ \frac{\log p(X^n|\hat{\theta}(X^n))^s}{\int p(X^n|\theta)^s q(\theta)d\theta} \right] - sD_s, \] (4)
where
\[ D_s = E_p[\log p(X^n|\hat{\theta}(X^n)) - \log p(X^n|\theta)] \]
is the average Gibbs training error relative to the MLE, and
\[ q_s(\theta|x^n) = \frac{p(x^n|\theta)^s q(\theta)}{\int p(x^n|\theta)^s q(\theta)d\theta}, \]
is the generalized posterior distribution, which has been used for purposes such as guaranteeing posterior consistency [5, 6] and model selection [7, 8, 9].

If the prior is chosen appropriately, for example, the uniform distribution on a compact subset of Θ including θ0, the following asymptotic expansion is obtained as n → ∞ and s → ∞ [10],
\[ E_p \left[ \frac{\log p(X^n|\hat{\theta}(X^n))^s}{\int p(X^n|\theta)^s q(\theta)d\theta} \right] \approx \lambda \log(ns) + o(\log(ns)), \]
(6)
where \( \hat{\theta}(x^n) \) is chosen from the subset of the parameter space on which the prior has positive probability. The coefficient \( \lambda \) is referred to as the learning coefficient, which depends on the model, prior and source. If the model is regular, \( 2\lambda = m \), the dimension of the parameter, whereas if it is non-regular, \( 2\lambda \leq m \) holds in general [10].

Substituting the asymptotic expressions of the Gibbs training error and the training error of the MLE [10, p. 159] into (5), we have
\[ D_s \simeq \nu + \left( \frac{\lambda}{s} - \nu \right) = \frac{\lambda}{s}, \] (7)
where \( \nu = \lim_{n \to \infty} D_{\max} \). Hence from (4), (6) and (7), we have
\[ R(D) \leq -\lambda \log D + o(\log D) \]
for this RD problem.

The above inequality implies that the RD dimension [11] defined by
\[ \dim_{RD} = \lim_{D \to 0} \frac{R(D)}{\frac{1}{2} \log D} \]
is upper bounded by 2\lambda, that is,
\[ \dim_{RD} \leq 2\lambda \]
holds and the learning coefficient provides an estimate (upper bound) of the RD dimension of this RD problem.

**Example 1:** If the source is the m-dimensional standard normal distribution with mean \( \theta_0 \in \mathbb{R}^m \) and the model is \( p(x|\theta) = N(x|\theta) \propto \exp \left( -\frac{1}{2} \|x - \theta\|^2 / 2 \right) \), the optimal reconstruction distribution \( q^*_n(\theta) \) is Gaussian and the RD function is
\[ R(D) = \frac{m}{2} \log \frac{m}{2D} \left( D \leq \frac{m}{2} \right), \]
which is essentially the same as the RD function of the Gaussian source under the squared error criterion obtained by Shannon [15], \( \dim_{RD} = m = 2\lambda \) holds if the prior is continuous.

**Example 2:** If the source is the Bernoulli distribution with success probability \( \theta_0 \) and the model is the Bernoulli distribution with the parameter \( \theta \), the RD function has endpoints \((0, R(0))\) and \((D_{\max}, 0)\), given by
\[ R(0) = -\sum_{k=0}^{n} p_k \log p_k, \]
\[ D_{\max} = n \left\{ h_2(\theta_0) = \sum_{k=0}^{n} p_k h_2 \left( \frac{k}{n} \right) \right\}, \]
where \( h_2 \) is the binary entropy function and \( p_k = \binom{n}{k} \theta_0^k (1 - \theta_0)^{n-k} \). Between these two endpoints, the optimal reconstruction distribution is given by a discrete distribution for each distortion level, which can be computed by an iterative algorithm [16]. The RD function for \( \theta_0 = 0.3 \) is illustrated in Fig. 1. The RD dimension of the RD function is zero whereas \( 2\lambda = 1 \) if the prior is continuous.
3.3. Shannon-Bregman Lower Bound

Let

\[ p_{\text{Bayes}}^{(s)}(x^n) = \phi_s(x^n) \int p(x^n|\theta)^s q(\theta) d\theta, \]

where \( \phi_s(x^n) \) is a function satisfying

\[ \int p(x^n|\theta)^s \phi_s(x^n) dx^n = 1 \]

for all \( \theta \). If such a function \( \phi_s(x^n) \) exists, we have the following lower bound to \( R(D) \),

\[
R(D) = \sup_{s \geq 0} q(\theta) \left\{ KL(p||p_{\text{Bayes}}^{(s)}) + E_p[\log \phi_s(X^n)] + h(p(x^n)) - sE_p[\log p(X^n|\hat{\theta}(X^n))] - sD \right\},
\]

which was named Shannon-Bregman lower bound in the case where \( p(x|\theta) \) is the exponential family model [17]. The lower bound is tight at the slope parameter \( s \) if and only if

\[ p(x^n) = p_{\text{Bayes}}^{(s)}(x^n) \quad (\forall x^n), \]

that is, the source is expressed as a Bayes mixture by some prior \( q(\theta) \). If the source is replaced with the normalized maximum likelihood distribution,

\[ p_{\text{NML}}(x^n) \propto p(x^n|\hat{\theta}(x^n)), \]

the tightness of the lower bound is equivalent to the representability of \( p_{\text{NML}} \) by \( p_{\text{Bayes}}^{(s)} \) for some \( q(\theta) \). The prior \( q(\theta) \) satisfying \( p_{\text{NML}} = p_{\text{Bayes}}^{(1)} \) has been studied [18].

4. DP-MEANS CLUSTERING

The Dirichlet process (DP) means method is a clustering algorithms which estimates the number of clusters from data. While this algorithm was derived as a limit of nonparametric Bayesian approach to Bayesian learning of mixture models [12, 13], it is a simple extension of the K-means algorithm. More specifically, DP-means runs the usual K-means with \( K = 1 \), and generates a new cluster when the distance from a data point to its nearest cluster center is larger than a constant \( \eta \), which is called the penalty parameter and prespecified by the user.

From an RD theoretic point of view, the penalty parameter can be considered as controlling the maximum distortion in the training data set \( \{x_1, \cdots, x_N\} \in R^d \) [19],

\[ \max_i d(x_i, \theta_{c(i)}), \]

where \( c(i) \) denotes the cluster label of the \( i \)th data point and \( \theta_1, \cdots, \theta_K \) are the cluster centers obtained by the algorithm. The logarithm of the estimated number of clusters per dimension of data, \( \log K / L \), can be considered as the rate. As well as the average distortion criterion, the RD theory of the maximum distortion criterion has been developed [20], and it was proved that if the source is i.i.d., the RD functions of the two criteria are identical [20]. This implies that

\[ \frac{\log K}{L} \rightarrow R(\eta) \]

as \( N \rightarrow \infty \) and \( L \rightarrow \infty \), where \( R \) is the RD function of the distortion measure used in the algorithm [19].

For example, if each data point \( x_i = \{x_i^{(1)}, \cdots, x_i^{(L)}\} \in \{0, 1, \cdots, n\}^L \) is generated from a binomial distribution, \( B(n, \theta_0) \), and the distortion measure \( d(x, \theta) \) is chosen to be

\[ \frac{1}{L} \sum_{j=1}^{L} \left\{ x^{(j)} \log \frac{x^{(j)}}{n\theta(\theta)} + (n - x^{(j)}) \log \frac{n - x^{(j)}}{n - n\theta(\theta)} \right\}, \]

then the limiting RD function is the same as the one discussed in Section 3 for the Bernoulli model (Fig. 1).

5. CONCLUSION

In this study, we discussed RD theoretic views of Bayesian learning and DP-means clustering. It is an interesting undertaking to investigate the RD functions under distortion measures corresponding to non-regular statistical models.

6. ACKNOWLEDGMENTS

The result of Section 4 is a joint work with Masahiro Kobayashi [19]. This work was supported in part by JSPS KAKENHI 25120014, 15K16050, and 16H02825.

7. REFERENCES


STOCHASTIC COMPLEXITY FOR LATENT VARIABLE MODELS

Kenji Yamanishi

The University of Tokyo,
7-3-1 Hongo, Bunkyo-ku, Tokyo, JAPAN
yamanishi@mist.i.u-tokyo.ac.jp

1. INTRODUCTION

Letting $X$ be an observed variable and $Z$ be a latent variable, we consider the probability distribution the density function of which is specified by a real-valued parameter vector $\theta$ and a discrete model $M$ as follows:

$$ p(X; \theta, M) = \sum_Z p(X, Z; \theta, M). $$

We call $p(X; \theta, M)$ a marginalized model while $p(X, Z; \theta, M)$ a complete variable model. We consider the model selection problem: For an observed sequence $x^n$, find the best model $M$ that explains it. The selection of the best model is equivalent with the best choice of the number of latent variables.

Conventional statistical model selection criteria, such as Akaike’s information criteria (AIC) and Bayesian information criteria (BIC) cannot straightforwardly be applied to such a latent variable model when the latent variables are marginalized out. Such a marginalized model is irregular in the sense that the model and parameters are not in one-to-one correspondence. For such an irregular model the central limit theorem (CLT) for the maximum likelihood estimator does not hold. The classical formula of AIC and BIC cannot be applied to irregular models since they are derived under the condition that CLT holds.

We apply the minimum description length (MDL) principle [1] not to the marginalized model but rather to the one for which the latent variables are completed. The model with the completed latent variables is no longer irregular. Let $\mathcal{L}(x^n, z^n; M)$ be the total code-length required for encoding an observable data sequence $x^n$ and a latent variable sequence $z^n$. We call $\mathcal{L}(x^n, z^n; M)$ the latent stochastic complexity, which we abbreviate as LSC. Then the MDL-based model selection for latent variable models is formulated as follows:

Given $x^n$, $\mathcal{L}(x^n, z^n; M) \Rightarrow \min \text{ w.r.t. } M$ and $z^n$.

It requires that LSC be minimized with respect to the latent variable sequence as well as the model. There are two problems in this model selection. One is an information-theoretic problem; how can we calculate LSC so that it is as short as possible? The other is a computational one; how efficiently can we compute LSC? We address these issues through the concrete classes of probabilistic models.

2. LATENT STOCHASTIC COMPLEXITY

We introduce two methods for calculating LSC. One is to employ the normalized maximum likelihood (NML) code-length. The other is to employ the two part code-length. Both methods are applied to the models with latent variable completed rather than those with latent variables marginalized out.

2.1. NML Code-length

2.1.1. Latent Parametric Complexity

The NML code-length is defined as the negative logarithm of the NML distribution. The NML distribution is a solution to the Shtarkov’s minimax regret as follows:

$$ p_{\text{NML}}(X^n, Z^n : M) = \arg \min_{\hat{\theta}} \max_{q(x^n, z^n)} \left\{ -\log q(x^n, z^n) - \min_{\theta} (-\log p(x^n, z^n; \theta, M)) \right\}, $$

where the minimum is taken over all probability distributions. Note that the minimax regret is defined relative to the complete variable model rather than the marginalized model.

For given $x^n$, the NML code-length is given by

$$ -\log p_{\text{NML}}(x^n, z^n : M) = -\log p(x^n, z^n; \hat{\theta}(x^n, z^n), M) + \sum_{w^n} \int dy^n p(y^n, w^n; \hat{\theta}(y^n, w^n), M), \quad (1) $$

This is because the models marginalized with respect to latent variables are irregular in general, in the sense that there is no one-to-one correspondence between parameters and distributions. In order to overcome this problem, this paper introduces the notion of latent stochastic complexity to propose how to select the best latent structures from given data. I show concrete methods for calculating latent stochastic complexity for basic classes of latent variable models.
where \( \hat{\theta}(x^n, z^n) \) is the maximum likelihood estimator of \( \theta \) from \( x^n \) and \( z^n \). The latent variable sequence \( z^n \) is not actually observed. Hence we select the best model \( M \) by minimizing the NML codeword with respect to \( z^n \) as well as \( M \). Since \( z^n \) depends only on the first term in (1), the estimate of \( z^n \) should be the maximum likelihood estimator for the complete variable model.

We call the second term in (1) the latent parametric complexity, which means the information-theoretic complexity of the complete model class. It takes a more complicated form than the ordinal parametric complexity for the marginal model in the sense that it requires the marginalization of the maximum likelihood with respect to \( z^n \) as well as \( x^n \). The critical issue is how precisely and efficiently we can calculate the latent parametric complexity for practical classes of latent variable models.

2.1.2. Mixture models

First let us consider the basic class, which we call finite mixture models. A combinatorial method works for efficient computation of latent stochastic complexity for this class. A probability density function for a finite mixture model is written as

\[
p(X; \theta, K) = \sum_{i=0}^{K} \pi_i p(X; \mu_i),
\]

where we let \( \theta = (\pi_0, \ldots, \pi_K, \mu_0, \ldots, \mu_K) \). This is the marginalized model. Let \( Z \) denote a latent variable indicating which component \( X \) is generated from. Then the complete variable model is given by

\[
p(X, Z = i; \theta) = \pi_i p(X|Z; \mu_i),
\]

where we let \( \pi_i = p(Z = i) \). We easily see that the marginalized model (2) is irregular. This is because there exist different \( \{\pi_i\} \) which may yield an identical probability density when \( \mu_i = \mu_j \) \((i \neq j)\). We are concerned with the issue of how we can select the best \( K \) from a given data sequence, on the basis of LSC.

The exponent of the latent stochastic complexity for this model is written as follows: Letting \( n_i \) be the number of examples assigned to the cluster of \( Z = i \),

\[
C_n(K) \defeq \sum_{\forall i, n_i \geq 0} \frac{n!}{n_0! \cdots n_K!} \prod_{i=0}^{K} \left( \frac{n_i}{n} \right)^{n_i} C_{n_i}, \quad (3)
\]

where \( C_{n_i} \) can be calculated using Rissanen’s asymptotic formula [2] for approximating the parametric complexity since the model for each component is no longer an irregular model:

\[
\log C_{n_i} = \frac{k}{2} \log \frac{n_i}{2\pi} + \log \int \sqrt{I(\theta)} d\theta + o(1), \quad (4)
\]

where \( k \) is the number of parameters and \( I(\theta) \) is the Fisher information matrix. The following theorem is useful for efficient computation of (3).

**Theorem 1** [3] The latent parametric complexity (3) for finite mixture models satisfies the recurrence relation:

\[
C_n(K + 1) = \sum_{r_1 + r_2 = n} \frac{n!}{r_1! r_2!} \left( \frac{r_1}{n} \right)^{r_1} \left( \frac{r_2}{n} \right)^{r_2} C_{r_1}(K) C_{r_2}.
\]

This relation enables us to compute the latent parametric complexity (3) in time \( O(n^2 K) \) for sample size \( n \).


2.1.3. Non-negative matrix factorization

Non-negative matrix factorization (NMF) is to factorize an \( N \times M \) non-negative matrix \( X \) into a product of two matrices with low rank as follows:

\[
X \approx Z \Theta;
\]

where \( \Theta = \{\theta_{km}\} \) is an \( N \times K \) real-valued non-negative matrix and \( Z = \{z_{nk}\} \) is a \( K \times M \) real-valued non-negative matrix. NMF can be thought of as clustering \( N \) data into \( K \) clusters where \( z_{nk} \) is the degree of membership of the \( n \)-th data into the \( k \)-th cluster.

According to [5], we may think of \( Z \) as a latent variable matrix and of \( \Theta \) as a parameter matrix. We define the latent variable model associated with NMF as follows:

\[
\begin{align*}
z_{nk} & \sim \text{G}(z_{nk}; \alpha_{nk}, \beta_{nk}), \\
s_{nkm} & \sim \text{Po}(s_{nkm}; z_{nk}\theta_{km}), \\
x_{nm} & = \sum_{k=1}^{K} s_{nkm},
\end{align*}
\]

where \( \text{G}(z; \alpha, \beta) \) is the Gamma distribution with density:

\[
z^{\alpha-1}e^{-z/\beta}/(\Gamma(\alpha)\beta^\alpha) \quad \text{and} \quad \text{Po}(s; \lambda) = \text{Poisson distribution with density:} \quad \lambda^s e^{-\lambda}/s!.
\]

This model for NMF is irregular since for all \( (n, m) \), there exist \( \{z_{nk}, \{\theta_{km}\}\} \neq \{z'_{nk}, \{\theta'_{km}\}\} \) such that

\[
\sum_k z_{nk}\theta_{km} = \sum_k z'_{nk}\theta'_{km}.
\]

It is an important issue how we can select the best \( K \) from given \( X \). We may select the best rank \( K \) so that it achieves the minimum of LSC:

\[
-\log p(X, Z; \Theta, \alpha, \beta) + \log C_{N,M}(K),
\]

where \( \hat{\Theta}, \hat{\alpha}, \hat{\beta} \) are maximum likelihood estimators of \( \Theta, \alpha, \beta \) and the second term is the latent parametric complexity. In this case an asymptotic method works for calculating the latent parametric complexity. Ito et al. proved that it is asymptotically expanded as follows:

**Theorem 2** [5] Supposing \( z_{nk} \in [z_{\text{min}}, z_{\text{max}}], \theta_{nm} \in (0, \theta_{\text{max}}) \), the latent parametric complexity for NMF is asymptotically given by
log $C_{N,M}(K)$

$$= MK \left( \frac{1}{2} \log \frac{N}{2\pi} + \log 2 \right) + K \log(z_{\text{max}} - z_{\text{min}})$$

$$+ \sum_{k=1}^{K} \log \int_{z_{\text{min}}}^{z_{\text{max}}} \exp \left( \frac{z_k \theta_{\text{max}}}{2} + \log \left( \frac{\log(z_k \theta_{\text{max}})}{\log 2} \right) \right) \, dz_k.$$ 


2.2. Two-part codelength

When LSC is calculated using the NML codelength, we may often suffer from the computational issue on latent parametric complexity for some model classes. For such cases, I propose to employ the two-part codelength in the calculation of LSC. In it, unlike the NML codelength, the observed variable sequence and the latent one are encoded separately. It is formally defined as follows:

$$\mathcal{L}(x^n, z^n : M) = \mathcal{L}_{\text{NML}}(x^n | z^n : M) + \mathcal{L}_{\text{NML}}(z^n : M),$$

where $\mathcal{L}_{\text{NML}}(x^n | z^n : M)$ is the NML codelength for $x^n$ conditioned for $z^n$ and $\mathcal{L}_{\text{NML}}(z^n : M)$ is the NML codelength for $z^n$ itself. We select the best model $M$ from given data $x^n$ so that the two-part codelength achieves the minimum with respect to $z^n$ as well as $M$. The two-part codelength is not optimal in the sense of Shtarkov’s minimax regret.

For the mixture model under the same notation as in Section 2.1.2, the two-part codelength may be calculated as follows: Letting $x^{n_i}$ be the data sequence assigned to the cluster $Z = i$,

$$\sum_{i=0}^{K} \left\{ - \log p(x^{n_i}; \hat{\theta}(x^{n_i})) + \sum_{y^{n_i}} p(y^{n_i}; \hat{\theta}(y^{n_i})) \right\}$$

$$- \log \prod_{i=0}^{K} \binom{n_i}{n}$$

$$+ \log \sum_{\forall i} n_i = n, n_i \geq 0$$

The best model is obtained by minimizing $z^n$ as well as the model. In (5), the first term is the NML code-length for $x^n$ conditioned for $z^n$ and the sum of the second and the third terms is the NML code-length for $z^n$ itself. The first term can be calculated using Rissanen’s asymptotic expansion formula (4). The third term can be efficiently computed using Kontkanen and Myllymäki’s algorithm for calculating the parametric complexity of multinomial distribution [10]. Hence through the two-part codelength we can overcome the computational problem on the latent parametric complexity for the NML codelength. This computational advantage is most effective in model selection for hierarchical topic models.

3. CONCLUSION

I have introduced the notion of latent stochastic complexity (LSC) in order to address the model selection issue for latent variable models. I have shown two methods for calculating LSC. One is the NML codelength for complete variable models. This is introduced in order to avoid the irregularity issue arising when latent variables are marginalized out. Several methods including a combinatorial method, an asymptotic one enable us to compute LSC efficiently for basic models such as mixture models, NMF, relational models, etc. The other is the two-part codelength for complete variable models. It has more computational advantages than the NML codelength. This works most effectively specifically when dealing with complicated latent variable models such as hierarchical topic models.

4. ACKNOWLEDGMENTS

This work was partially supported by JST-CREST.

5. REFERENCES


