Decomposed Normalized Maximum Likelihood Codelength Criterion for Selecting Hierarchical Latent Variable Models

Tianyi Wu
Graduate School of Information Science and Technology
The University of Tokyo
Tokyo, JAPAN
tianyi_wu@mist.i.u-tokyo.ac.jp

Shinya Sugawara
Graduate School of Information Science and Technology
The University of Tokyo
Tokyo, JAPAN
sugawara_shinya@ci.i.u-tokyo.ac.jp

Kenji Yamanishi
Graduate School of Information Science and Technology,
The University of Tokyo
Tokyo, JAPAN
yamanishi@mist.i.u-tokyo.ac.jp

ABSTRACT

We propose a new model selection criterion based on the minimum description length principle in a name of the *decomposed normalized maximum likelihood* criterion. Our criterion can be applied to a large class of hierarchical latent variable models, such as the Naïve Bayes models, stochastic block models and latent Dirichlet allocations, for which many conventional information criteria cannot be straightforwardly applied due to irregularity of latent variable models. Our method also has an advantage that it can be exactly evaluated without asymptotic approximation with small time complexity. Our experiments using synthetic and real data demonstrated validity of our method in terms of computational efficiency and model selection accuracy, while our criterion especially dominated the other criteria when sample size is small and when data are noisy.

CCS CONCEPTS

• **Mathematics of computing** →**Coding theory;** • **Theory of computation** →*Unsupervised learning and clustering;* • **Computing methodologies** →**Latent variable models;**

KEYWORDS

Model selection; Hierarchical latent variable models; MDL

1 INTRODUCTION

1.1 Motivation

We are concerned with the issue of selecting the best probabilistic model from a data sequence for a class of hierarchical latent variable models. Letting X be an observed variable and Z be a latent variable, we consider the probability mass function (or density function) which is specified by a real-valued parameter vector θ and a discrete model M as follows:

$$P(X;\theta,M)=\sum_{Z} P(X,Z;\theta,M).$$

Here M is, for example, the number of latent variables, clustering structures, etc. We call $P(X;\theta,M)$ a *marginalized model* while $P(X,Z;\theta,M)$ a *complete variable model*, either of which is referred to as a *latent variable model*. We consider the model selection problem: For an observed sequence $x^n = x_1,\ldots,x_n$, find the best model M that explains it.

Conventional statistical model selection criteria, such as Akaike's information criteria (AIC) [2], Bayesian information criteria (BIC) [18], etc. 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. In an irregular model, the central limit theorem (CLT) for the maximum likelihood estimator does not hold. Since AIC, BIC are derived under the condition that CLT holds, they do not have a theoretical foundation to work for marginalized models

One of the ways to overcome this irregularity problem is to apply the *minimum description length* (MDL) principle [14] not to the marginalized model but rather to the one for which the latent variables are completed. The MDL principle asserts from the information-theoretic view that the best model should be the one for which the total codelength is minimum. Let $\mathcal{L}(x^n,z^n;M)$ be the total codelength required for encoding an observable data sequence $x^n = x_1,\ldots,x_n$ and a latent variable sequence $z^n = z_1,\ldots,z_n$. Then the MDL-based model selection is formulated as follows:

$$\text{Given } x^n,\quad \mathcal{L}(x^n,z^n;M) \Longrightarrow \min \text{ w.r.t. } M \text{ and } z^n. \qquad (1)$$

Specifically, we may compute the *normalized maximum likelihood* (NML) codelength [14] as follows:

$$\mathcal{L}_{\mathrm{NML}}(x^n,z^n;M) \stackrel{\mathrm{def}}{=} -\log P(x^n,z^n;\hat{\theta}(x^n,z^n),M)+\log C(n,M),$$

where $\hat{\theta}(x^n,z^n)$ is the maximum likelihood estimator of θ given x^n and z^n, and $C(n,M)$ is the normalization term:

$$C(n,M) \stackrel{\mathrm{def}}{=} \sum_{x^n}\sum_{z^n} P(x^n,z^n;\hat{\theta}(x^n,z^n),M). \qquad (2)$$

Here $C(n,M)$ is called the *latent parametric complexity* (LPC), which measures the complexity of the model.

The reason why the NML codelength is employed is that it provides a unique solution to Shtarkov's minimax risk [19].

KDD'17, August 13–17, 2017, Halifax, NS, Canada.
 ISBN 978-1-4503-4887-4/17/08…$15.00
DOI: http://dx.doi.org/10.1145/3097983.3098110

According to the MDL principle, the best model is selected considering the trade-off between the goodness of fit to given data and the complexity of the model.

Although several studies proved that the MDL-based model selection works well to select the number of latent variables [7, 11], there remains a critical issue that the computation of the normalization term (2) is intractable for most of important latent variable models such as topic models and relational models. Actually, exact evaluation for the NML codelength is given only to few models and naïve computation for NML, if possible, takes an exponential time. Besides, although an asymptotic approximation of (2) has been proposed in [15], it cannot achieve good approximation of NML for small sample size and the approximation itself is difficult to compute for complicated latent variable models.

The purpose of this paper is twofold. The first is to develop a novel MDL-based model selection criterion in order to overcome the computational difficulty of the NML codelength. The second is to theoretically and empirically demonstrate the effectiveness of our proposed criterion for a number of important latent variable models both in terms of computational efficiency and model selection accuracy.

1.2 Contribution of This Paper

The contribution of this paper is summarized as follows:

1) *Proposal of decomposed normalized maximum likelihood (DNML) codelength criterion:* To overcome the computational difficulty of the NML codelength, we propose a novel MDL-based model selection criterion for latent variable models, which we call the *decomposed normalized maximum likelihood* (DNML) codelength criterion. It is considered as an approximation of the exact NML codelength. The key idea of DNML is to decompose the model into two parts and then to encode them separately with respect to models for latent variables and the model for observed data given latent variables. Our DNML criterion can be evaluated as a non-asymptotic value. We show that the DNML codelength is a good approximation of the NML codelength both using the asymptotic theory and numerical experiments. We also show that our criterion is efficiently computable. The total time complexity is only the time complexity for the NML codelength of latent components plus an additional linear time with respect to the data size. In hierarchical latent variable models, the latent components often follow simple distributions and their NML codelength can be efficiently computed in linear time. DNML gives a new computational insight of the information-theoretic model selection criteria.

2) *Applicability to a wide family of hierarchical latent variable models:* To show the validity of DNML, we apply it to the issue of selecting the number of latent components in several important models with latent variables. We derive closed-form expressions of non-asymptotic values of the DNML codelengths for the Naïve Bayes models (NB), stochastic block model (SBM) [20], mixed membership models including latent Dirichlet allocation (LDA) [5], and mixed membership SBM models [1], etc. They are all considered as *hierarchical latent variable models*, in each of which latent variables and their hyper parameters form a hierarchical structure. Time complexities for computation of DNML are shown to be linear in sample sizes for these models. Note that for all of them, it is intractable to compute the exact values of the NML codelength, or it is not easy to calculate its tight approximated value. Figure 1 shows relations between hierarchical latent variable models for which DNML is effective. It shows wide applicability of DNML into model selection for important models.

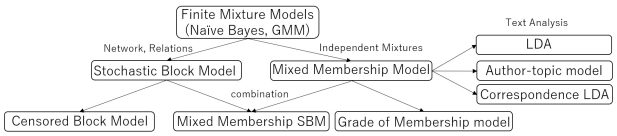

Figure 1: A family of hierarchical latent variable models to which DNML is applicable

3) *Empirical demonstration of the effectiveness of DNML:* We conduct experimental studies using synthetic datasets and benchmark real datasets to compare our criterion with conventional model selection criteria, such as AIC, BIC, approximated marginal likelihood via the Laplace method or via the variational Bayes bound, non-parametric Bayes and clustering criteria. Experimental results demonstrate that DNML selected the true number of latent components appropriately in general. Especially, DNML showed faster convergence to the true number of components than the other criteria. Furthermore, an experiment for noisy simulation data shows that DNML performs better than the other criterion including AIC, which suffers from over-fitting under noise.

1.3 Related Work

Model selection criteria for latent variable models have been explored recently. On the basis of the MDL principle, the NML codelength for the completed variable models have been used for model selection of Naïve Bayes models [11], Gaussian mixture models [7], and a simplified variant of SBM [17], and non-negative matrix factorization [8]. In most of these studies, the NML codelength criterion is computed using the Rissanen's asymptotic approximation formula [15]. However, it does not always give a good approximation to the true NML codelength for small data. Further, NML suffers from the computational issue when calculating LPC for complicated models. Hence the methodologies of the previous work cannot straightforwardly be applied into a family of hierarchical latent variable models.

Recently, nonparametric Bayesian methods have gained a popularity in researches for latent variable models, because they can estimate a model as if it is also a parameter. In other words, these methods are accompanied with a genuine model selection procudure. For LDA, the hierarchical Dirichlet process (HDP) [22] can be applied to the selection of the number of topics. For SBM, infinite relational models [9] can be applied to the selection of the numbers of blocks. We compare all of them with DNML in Section 4.

Source codes for this research are available from a Github repository https://github.com/tianyi-wu/DNML. Proofs for theorems and detailed designs of experiments are provided in the supplemental materials [24].

2 THE DNML CRITERION

2.1 Model Selection via NML Codelength

Our data consist of a sequence of n observations $x^n = x_1, \ldots, x_n$. In this research, we concentrate on models with latent variables. In other words, we consider a corresponding sequence of latent variable $z^n = z_1, \ldots, z_n$ and construct candidates of a joint probabilistic model for data and latent variables as $P(x^n, z^n; \theta, M)$ where θ is a vector of unknown parameters and $M \in \mathcal{M}$ indexes a model among a set of candidate models. A problem we face is how to choose the best model among the candidates. We take the model selection strategy of (1) on the basis of the MDL principle.

In this study, we focus on a class of models with discrete latent variables such that there are K latent clusters to which each of data belong. Further, the discrete latent variable z^n are assumed to represent an index for a cluster membership and each cluster is accompanied with a parameter vector θ. A conventional approach to this model selection problem of (1) is using the *normalized maximum likelihood* (NML) codelength [13] (or simply, NML), defined as:

$$\begin{aligned}\mathcal{L}_{\mathrm{NML}}(x^n, z^n; M) &\stackrel{\mathrm{def}}{=} -\log P_{\mathrm{NML}}(x^n, z^n; M) \\ &= -\log \frac{P(x^n, z^n; \hat{\theta}(x^n, z^n), M)}{C(n, M)},\end{aligned}$$

where $P_{\mathrm{NML}}(x^n, z^n; M)$ is the NML distribution, $\hat{\theta}(x^n, z^n)$ is the maximum likelihood estimator of θ from x^n and z^n and the denominator is a normalization constant, which is called *latent parametric complexity* (LPC), defined as:

$$C(n, M) \stackrel{\mathrm{def}}{=} \sum_{x^n} \sum_{z^n} P(x^n, z^n; \hat{\theta}(x^n, z^n), M). \quad (3)$$

LPC represents the information-theoretic complexity for a class of complete variable models. In general, it is difficult to obtain the exact value of the NML code-length because the summation in LPC as (3) is taken over the set of all data sequences of length n.

Instead of exact evaluation, Rissanen [15] proposes an asymptotic approximation formula for LPC as follows:

$$\log C(n, M) = \frac{K}{2} \log \frac{n}{2\pi} + \log \int \sqrt{|I(\theta)|} d\theta + o(1), \quad (4)$$

where K is the number of independent parameters, $I(\theta)$ is the Fisher information matrix and $o(1)$ means $\lim_{n\to\infty} o(1) = 0$ uniformly over all data sequences. As shown in Section 4, this asymptotic approximation does not work well for small sample sizes. Further, $\int \sqrt{|I(\theta)|} d\theta$ is difficult to compute for many hierarchical latent variable models of our concern.

2.2 Definition of DNML

We introduce the *decomposed normalized maximum likelihood* (DNML) codelength. It encodes the observed data and latent variables separately via the NML codelength as:

$$\mathcal{L}_{\mathrm{DNML}}(x^n, z^n; M) \stackrel{\mathrm{def}}{=} \mathcal{L}_{\mathrm{NML}}(x^n | z^n; M) + \mathcal{L}_{\mathrm{NML}}(z^n; M), \quad (5)$$

where

$$\begin{aligned}\mathcal{L}_{\mathrm{NML}}(x^n | z^n; M) &\stackrel{\mathrm{def}}{=} -\log P_{\mathrm{NML}}(x^n | z^n; M) \\ &= -\log P(x^n | z^n; \hat{\theta}(x^n, z^n), M) \\ &\quad + \log \sum_{x'^n} P(x'^n | z^n; \hat{\theta}(x'^n, z^n), M), \quad (6)\end{aligned}$$

$$\begin{aligned}\mathcal{L}_{\mathrm{NML}}(z^n; M) &\stackrel{\mathrm{def}}{=} -\log P_{\mathrm{NML}}(z^n; M) \\ &= -\log P(z^n; \hat{\theta}(z^n), M) \\ &\quad + \log \sum_{z'^n} P(z'^n; \hat{\theta}(z'^n), M). \quad (7)\end{aligned}$$

We call (5) as the DNML codelength (or simply, DNML). The DNML criterion is to select M and z^n so that DNML is minimized with respect to M and z^n for given x^n.

In the NML codelength, observed data x^n and latent variables z^n are encoded simultaneously. The main difficulty comes from the computation of LPC (3) which needs to evaluate the joint distribution function for data and latent variables along with a long sequence to calculate a summation term. Instead, in this proposed DNML codelength, we can avoid the computation problem by encoding x^n and z^n separately into two parts.

The codelengths in (6) and (7) can be further simplified for hierarchical latent variable models. For (6), given z^n, the joint likelihood of x^n can be expressed as the product of individual likelihood functions of each latent component. Letting x_k^n, z_k^n be the data and the latent variables that belong to the kth latent component given z^n, we have

$$\begin{aligned}\mathcal{L}_{\mathrm{NML}}(x^n | z^n; M) &= -\log \prod_k P(x_k^n | z_k^n; \hat{\theta}(x_k^n, z_k^n), M) \\ &\quad + \log \prod_k \sum_{{x'_k}^n} P({x'_k}^n | z_k^n; \hat{\theta}({x'_k}^n, z_k^n), M) \\ &= \sum_k \mathcal{L}_{\mathrm{NML}}(x_k^n | z_k^n; M). \quad (8)\end{aligned}$$

As seen in the next section, the exact value of $\mathcal{L}_{\mathrm{NML}}(x^n | z^n; M)$ can be calculated in linear time in n.

In (7), for finite mixture models with K mixtures, $P(z^n; \theta, M)$ is just a probability mass function for the multinomial distribution. We let n_k be the number of data in the kth latent component. Then, $\mathcal{L}_{\mathrm{NML}}(z^n; M)$ can be computed as follows:

$$\mathcal{L}_{\mathrm{NML}}(z^n; M) = -\log \prod_{k=1}^{K} \left(\frac{n_k}{n}\right)^{n_k} + \log C_{\mathrm{MN}}(n, K), \quad (9)$$

where $C_{\mathrm{MN}}(n, K) = \sum_{z^n} \prod_{k=1}^{K} (n_k/n)^{n_k}$ is LPC for multinomial distributions. Kontkanen and Myllymäki [10] showed that $C_{\mathrm{MN}}(n, K)$ can be computed in linear time with respect to n using the recurrence relation:

$$C_{\mathrm{MN}}(n, K) = C_{\mathrm{MN}}(n, K-1) + \frac{n}{K-2} C_{\mathrm{MN}}(n, K-2). \quad (10)$$

For more complicated models like the mixture membership model with D groups, $P(z^n;\theta,M)$ is a product of multinomial distributions for each distinct group. If we let z^n_d be an index that means a corresponding observation belong to group d, (7) can be written as $\sum_d \mathcal{L}_{\mathrm{NML}}(z^n_d;M)$.

2.3 Approximation Accuracy of DNML

Let us validate how well our proposed DNML can approximate the exact NML. We denote K_{total} as the total number of independent parameters in the model, $K_{mixture}$ as the number of independent parameters in $P(z^n;\theta)$ and K^k_{base} as the number of independent parameters in the kth latent component. For finite mixture models, $K_{mixture}=K-1$ and the following theorem shows that the DNML codelength approximates the NML codelength well.

THEOREM 2.1. *Let $c_k=n_k/n$ for finite mixture models where n_k is the number of data belonging to a cluster k given z^n. Assume that $n_k=\Theta(n)$ for all k. As $n\to\infty$, we obtain*

$$\begin{aligned}&\mathcal{L}_{\mathrm{NML}}(x^n,z^n;M)-\mathcal{L}_{\mathrm{DNML}}(x^n,z^n;M)\\ &=-\log\Gamma\left(\frac{\sum_k K^k_{base}+K}{2}\right)+\log\Gamma\left(\frac{K}{2}\right)-\frac{K}{2}\log\pi\\ &\quad+\sum_k\left\{\log\Gamma\left(\frac{K^k_{base}+1}{2}\right)-\frac{K^k_{base}}{2}\log c_k\right\}+o(1).\end{aligned}\tag{11}$$

In the specific case where data are equally divided, i.e. $c_k=1/K$, (11) becomes

$$\mathcal{L}_{\mathrm{NML}}(x^n,z^n;M)-\mathcal{L}_{\mathrm{DNML}}(x^n,z^n;M)\approx\frac{K}{2}\log\frac{K}{2\pi e}.\tag{12}$$

See Appendix for proof. Note that (12) is a relatively small value. For example, (12)$=0.789$ for $K=10$. Theorem 2.1 implies that DNML is a good approximation of NML, hence we may employ DNML in place of NML provided that DNML can be more efficiently computed than NML. The condition in Theorem 2.1 implies that there does not exist very small clusters when n is large. Even if this condition does not hold, we can find small clusters and remove them in advance for many cases.

3 DNML FOR HIERARCHICAL LATENT VARIABLE MODELS

3.1 Naïve Bayes models

NB is one of finite mixture models. Although we focus on categorical data, extension to continuous data can be done following [7]. Let K be the number of latent components, $L=(L_1,...,L_D)$ be the dimensions of an observed data point $x_i=(x_{i1},..,x_{iD})$, π be the mixture probability and $\phi_k=(\phi_{k1},...,\phi_{kD})$ be the probability of observed data where $\phi_{kd}=(\phi_{kd1},...,\phi_{kdL_d})$, $\sum_{l=1}^{L_d}\phi_{kdl}=1$ is the probability for the d-th dimension. The generative process is given as:

- For observations $i=1,...,n$:
  (1) Generate a latent variable $z_i\sim\mathrm{Multi}(\pi)$.
  (2) For dimension $d=1,...,D$:
    – Generate $x_{id}\sim\mathrm{Multi}(\phi_{z_id})$ for data x_i.

Let n_{kdl} be the number of occurrences of l in dimension d of latent component k, n_{kd} be the number of data in dimension d of latent component k and n_k be the number of data in latent component k. The following theorem shows that (5) can be efficiently computed for NB.

THEOREM 3.1. *The DNML codelength for NB, $\mathcal{L}^{\mathrm{NB}}_{\mathrm{DNML}}(x^n,z^n;M)$, is given by*

$$\begin{aligned}&\mathcal{L}^{\mathrm{NB}}_{\mathrm{DNML}}(x^n,z^n;M)\\ &=\sum_k\sum_d\sum_l n_{kdl}\left(\log n_{kd}-\log n_{kdl}\right)+\sum_k\sum_d C_{\mathrm{MN}}(n_{kd},L_d)\\ &\quad+\sum_k n_k\left(\log n-\log n_k\right)+\log C_{\mathrm{MN}}(n,K),\end{aligned}$$

which is computed in time $O(n+K)$ where $n=\sum_{k,d}n_{kd}$. $C_{\mathrm{MN}}(n,K)$ is as in (10).

Proof sketch: We begin with deriving the expression for (6). Note that when latent variables z^n are given, the conditional maximum likelihood $P(x^n|z^n;\hat\Phi(x^n,z^n))$ can be written as follows:

$$P(x^n|z^n;\hat\Phi(x^n,z^n))=\prod_k\prod_d\prod_l\left(\frac{n_{kdl}}{n_{kd}}\right)^{n_{kdl}}.$$

Taking its negative logarithm, we get the first term in the main equation of Theorem. The second term represents the logarithm of the parametric complexity of $P(x^n|z^n;\hat\Phi)$ and can be computed as follows:

$$\begin{aligned}\sum_{x^n}P(x^n|z^n;\hat\Phi)&=\sum_{x^n}\prod_k\prod_d\prod_l\left(\frac{n_{kdl}}{n_{kd}}\right)^{n_{kdl}}\\ &=\prod_k\prod_d\sum_{x^n_{kd}}\prod_l\left(\frac{n_{kdl}}{n_{kd}}\right)^{n_{kdl}}\\ &=\prod_k\prod_d C_{MN}(n_k,L_d).\end{aligned}$$

For (7), because NB is a finite mixture model, the last two terms in the main equation of Theorem is derived from (9). For the time complexity, since n_{kd},n_k can be computed via a single pass through data and $C_{\mathrm{MN}}(n_k,L_d)$ can be computed in linear time [10], the total time complexity is linear. □

3.2 Stochastic Block Models

SBM [20] is a canonical model for community detection which partitions the vertices of a network into clusters. The model assumes that every cluster has its own probability to generate a link or not. Specifically, we let π be the mixture probability for K clusters and $\eta_{k_1k_2}$ be the probability of link density between cluster k_1 and k_2. The generative process can be described as follows:

(1) For vertex $i=1,...,n$:
  - Generate a latent variable $z_i\sim\mathrm{Multi}(\pi)$.
(2) For vertex $i_1=1,...,n$:
  - For vertex $i_2=1,...,n$:
    – Generate a variable $x_{i_1i_2}\sim\mathrm{Ber}(\eta_{z_{i_1}z_{i_2}})$.

The next theorem shows an efficient computation for (5).

THEOREM 3.2. *The DNML codelength* $\mathcal{L}^{\mathrm{SBM}}_{\mathrm{DNML}}(x^n, z^n; M)$ *for SBM is calculated as follows:*

$$\begin{aligned}&\mathcal{L}^{\mathrm{SBM}}_{\mathrm{DNML}}(x^n, z^n; M)\\ &= \sum_{k_1}\sum_{k_2}\left(n_{k_1k_2}\log n_{k_1k_2} - n^1_{k_1k_2}\log n^1_{k_1k_2} - n^0_{k_1k_2}\log n^0_{k_1k_2}\right)\\ &\quad + \sum_{k_1}\sum_{k_2}\log C_{\mathrm{MN}}(n_{k_1k_2}, 2)\\ &\quad + \sum_k n_k\left(\log n - \log n_k\right) + \log C_{\mathrm{MN}}(n, K),\end{aligned}$$

where $n^1_{k_1k_2}$ *and* $n^0_{k_1k_2}$ *are the number of links and no-links in cluster* (k_1, k_2) *and* $n_{k_1k_2}$ *is the total number of occurrences in cluster* (k_1, k_2)*. It is computable in time* $O(n + K)$*.*

3.3 Latent Dirichlet Allocations

LDA [5] is a model to analyze large collections of text data. For an LDA model with K topics, D documents and V unique terms, we denote θ_d as the topic mixture parameter of document d, ϕ_k as the word mixture parameter of topic k and α and β be the hyper-parameters for topic distributions θ_d and word distributions ϕ_k, respectively. The generative process of LDA can be describe as follows:

(1) For topic $k = 1, ..., K$:
  - Generate a word distribution $\phi_k \sim \mathrm{Dir}(\beta)$.

(2) For document $d = 1, ..., D$:
  (a) Generate a topic mixture $\theta_d \sim \mathrm{Dir}(\alpha)$.
  (b) For word $i = 1, ..., n_d$ in document d:
    (i) Generate a latent variable $z_{di} \sim \mathrm{Multi}(\theta_d)$.
    (ii) Generate an observed variable $x_{di} \sim \mathrm{Multi}(\phi_{z_{di}})$.

The following theorem gives a closed-form expression of DNML for LDA and shows that it is efficiently computable.

THEOREM 3.3. *The DNML codelength* $\mathcal{L}^{\mathrm{LDA}}_{\mathrm{DNML}}(x^n, z^n; M)$ *for LDA is given as follows:*

$$\begin{aligned}&\mathcal{L}^{\mathrm{LDA}}_{\mathrm{DNML}}(x^n, z^n; M)\\ &= \sum_k\sum_v n_{kv}\left(\log n_k - \log n_{kv}\right) + \sum_k \log C_{\mathrm{MN}}(n_k, V)\\ &\quad + \sum_d\sum_k n_{dk}\left(\log n_d - \log n_{dk}\right) + \sum_d \log C_{\mathrm{MN}}(n_d, K),\end{aligned}$$

where n_{kv} *is the number of word* v *in topic* k*,* n_k *is the total number of words in topic* k*,* n_{dk} *is the number of words in topic* k *from document* d *and* n_d *is the total number of words in document* d*. The DNML codelength for LDA is computed in time* $O(n + K + V)$ *where* $n = \sum_k n_k$*.*

3.4 Mixed Membership Stochastic Block Models

One limitation of SBM is that each data can only belong to one cluster. Airoldi et al. [1] relaxed this assumption and developed MMSBM for relational data. Let K be the number of clusters, α be the hyper-parameter for cluster distributions and $\eta_{k_1k_2}$ be the probability of link density between cluster k_1 and k_2, the generative process is:

(1) For vertex $i = 1, ..., n$:
  - Generate the topic mixture $\theta_i \sim \mathrm{Dir}(\alpha)$.

(2) For vertex $i_1 = 1, ..., n$: For vertex $i_2 = 1, ..., n$:
  (a) Generate a latent variable $z^{i_2}_{i_1} \sim \mathrm{Multi}(\theta_{i_1})$.
  (b) Generate a latent variable $z^{i_1}_{i_2} \sim \mathrm{Multi}(\theta_{i_2})$.
  (c) Generate an observed variable $x_{i_1i_2} \sim \mathrm{Ber}(\eta_{z^{i_2}_{i_1} z^{i_1}_{i_2}})$.

An efficient computation for DNML is given by the following theorem.

THEOREM 3.4. *The DNML codelength* $\mathcal{L}^{\mathrm{MMSBM}}_{\mathrm{DNML}}(x^n, z^n; M)$ *for MMSBM is calculated as follows:*

$$\begin{aligned}&\mathcal{L}^{\mathrm{MMSBM}}_{\mathrm{DNML}}(x^n, z^n; M)\\ &= \sum_{k_1}\sum_{k_2}\left(n_{k_1k_2}\log n_{k_1k_2} - n^1_{k_1k_2}\log n^1_{k_1k_2} - n^0_{k_1k_2}\log n^0_{k_1k_2}\right)\\ &\quad + \sum_{k_1}\sum_{k_2}\log C_{\mathrm{MN}}(n_{k_1k_2}, 2)\\ &\quad + \sum_i\sum_k n_{ik}\left(\log n_i - \log n_{ik}\right) + \sum_i \log C_{\mathrm{MN}}(n_i, K),\end{aligned}$$

where $n^1_{k_1k_2}$ *and* $n^0_{k_1k_2}$ *are the number of occurrences of links and no-links in cluster* (k_1, k_2)*,* $n_{k_1k_2}$ *is the total number of occurrences in cluster* (k_1, k_2)*.* n_{vk} *are the number of occurrences in cluster* k *from vertex* v *and* n_v *are the number of occurrences from vertex* v*. The DNML codelength for MMSBM is computed in time* $O(n + K)$ *where* $n = \sum_v n_v$*.*

3.5 Other Latent Variable Models

We can apply DNML to other hierarchical latent variable models as far as they have clustering structures.

- Author-topic model [16]: $P(x^n|z^n)$ of this model is the same as that of LDA. However, z^n do not follow the multinomial distribution but follow a two-level multinomial distribution. We can still efficiently calculate DNML using the recurrence relation in [11].
- Correspondence LDA [4]: This model is designed to model multi-dimensional data, while $P(x^n|z^n)$ and $P(z^n)$ can still be decomposed into products of each dimension. Therefore, we can simply compute the total DNML codelength as the sum of DNML codelengths for each dimension.

Furthermore, the exact expressions for models with continuous observed variables can be obtained by replacing the discrete observed variables with variables from Gaussian distributions or other continuous distributions. We only need to modify $C_{\mathrm{MN}}(n, K)$ to other simple parametric complexities which can be efficiently computed, as shown by Hirai and Yamanishi [7] for the Gaussian distribution.

4 EXPERIMENTS: SYNTHETIC DATA

In this section, we conduct five experiments using synthetic data. The first experiment considers whether DNML provides a good approximation of NML. The other four experiments compare DNML with conventional model selection methods.

To demonstrate applicability of DNML for a wide class of latent mixture models, the experiments handles distinct probabilistic models. The first and second experiments consider NB, the third experiment analyzes SBM. The fourth and fifth

experiments study LDA, for noiseless and noisy environments, respectively. For all models, in order to show robustness of DNML under diverse settings, we generate multiple artificial datasets using various values of hyper-parameters.

The proposed DNML has an advantage that it does not require an asymptotic theory in its justification and thus can work with a small sample size. Thus, we conducted experiments to see how accurately DNML is able to detect the right number of clusters even when it is relatively small. When the number of clusters might be considerably large, it does not make sense to consider which would be better to select, say, 1000 or 1001 clusters.

4.1 Experiment 1: Comparison of DNML and NML: NB

We investigate whether the DNML codelength is a good approximation of the exact NML. To this end, we conduct an experiment to generate artificial datasets according to NB, for which the exact NML codelength can be computed in $O(n^2 \log K)$ time [11]. Then, we directly compare DNML with the exact NML to evaluate the performance of approximation. We also derive Rissanen's approximation (4) (approximated NML, abbreviated as a-NML) for NB.

To guarantee the generality of experiments, we generated eight datasets for each combination of hyper-parameters from 480 candidates. We set the true number of latent components $K_{true} = 5$ and evaluated the codelengths for candidates with 2 to 8 latent components.

Table 1 gives the average ratios of the codelengths for DNML and a-NML. The term "ratio" means that the codelength is divided by the exact NML. Thus, a value close to one indicates a good approximation.

In this experiment, a-NML exhibited poorer performance than DNML for small n. This indicates the advantage of our DNML over a-NML. Also, when the assumption in the Theorem 2.1 held true, we found a significantly good performance of approximation. For models with $K \in \{3, 4, 5\}$, the numbers of selected latent components were at most the same as the true number 5. Thus the assumption: $n_k = \Theta(n)$ for all k in Theorem 2.1 holds true when n becomes large. Table 1 actually showed a good performance of DNML for $K = 3$ and 5. On the other hand, when $K \in \{6, 7\}$, the assumption in Theorem 2.1 does not hold for small n because there are too many latent components to model the data. Table 1 actually showed a relatively poor performance of DNML for $K = 7$ with small n.

This result corresponds to the statement of Theorem 2.1, which shows that DNML can be a good approximation of NML if clusters are evenly distributed. In this experiment, we showed the effectiveness of DNML with smaller K and larger n when clusters are more likely to be distributed evenly.

4.2 Experiment 2: Model Selection: NB

Next, we show simulation results on model selection for NB. The design of experiments is the same as with Section 4.1.

We compared DNML with conventional model selection methods: Rissanen's approximation for NML (a-NML) and exact NML (exact-NML), AIC with the latent variables completion (AIC), BIC with the latent variables completion (BIC), cross validation (CV), entropy of learned clustering (Entropy) and purity of learned clustering (Purity).

Table 1: Average ratios of codelengths via DNML and a-NML divided by the exact NML

No. of cluster K \ Sample size n					
DNML	**10**	**37**	**138**	**268**	**1000**
3	0.998	1.005	1.001	1.001	1.000
5	0.939	1.001	0.998	0.998	0.999
7	0.887	0.990	0.993	0.994	0.997
a-NML	**10**	**37**	**138**	**268**	**1000**
3	0.416	0.897	0.983	0.993	0.999
5	-0.105	0.790	0.963	0.985	0.998
7	-0.645	0.675	0.942	0.977	0.996

Due to irregularity explained in Section 1.1, AIC and BIC cannot straightforwardly be applied. Here we employ the complete model variants of AIC and BIC, where latent variables z^n are sampled from posterior distributions calculated by the EM algorithm. After completing z^n, the maximum likelihood estimation can be easily obtained and we consider the model without prior distributions on parameters. Entropy and purity are originally designed to evaluate performance of a clustering analysis. Purity is defined as $(1/n)\sum_{q=1}^{k} \max_{1\le j\le l} n(q,j)$, where n is sample size and $n(q,j)$ is the number of samples in cluster q that belongs to label $j (1 \le j \le l)$. Entropy is defined as $-(1/n)\sum_q \sum_j n(q,j)\log[n(q,j)/n_q]$, where n_q is the total number of samples in cluster q.

Figure 2 shows the graph of the selected number of latent components. DNML showed the best performance similar to the exact NML. a-NML (Rissanen's approximation) and AIC showed slower convergence, and BIC showed much more slower convergence, but all of these criteria yielded convergence to the true model ($K = 5$). Purity, Entropy and CV failed to selected the true model for all n. As a result, DNML has an advantage over the other model selection methods, especially for small sample sizes.

4.3 Experiment 3: Model Selection: SBM

We analyze the model selection for SBM. As for a-NML for SBM, we employed the result according to [17]. As for methods for comparison, in addition to AIC, BIC, Entropy and Purity, we employed the integrated classification likelihood (ICL) and the infinite relational model (IRM).

ICL[6] calculates the integrated log likelihood for the complete variable model based on the Laplace and the Stirling approximations. IRM [9] is a nonparametric Bayesian method which samples the number K during inference. We utilized the C source code by the author [1] with adopting the hyper-parameters in the code.

[1] http://www.psy.cmu.edu/ ckemp/code/irm.html, accessed Feb. 2, 2017

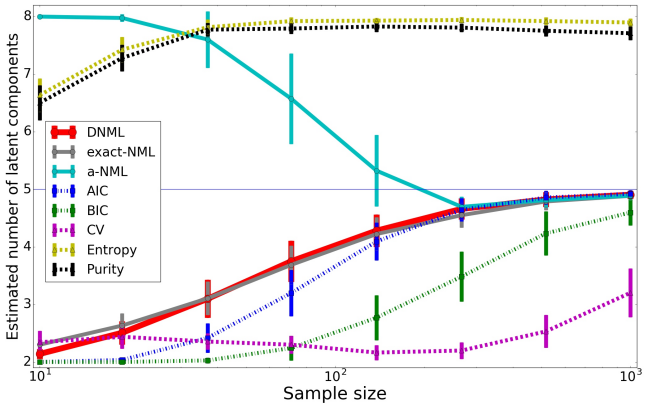

Figure 2: NB model: Estimated number K vs sample size n with $K_{true} = 5$

The hyper-parameters are α, β and ρ where $\pi \sim \mathrm{Dir}(\alpha)$, $\eta_{k_1 k_2} \sim \mathrm{Ber}(\beta)$. We generated eight datasets for each combination of hyper-parameters from 200 candidates. We set the true number of latent components to $K_{true} = 5$ and the best model was selected from candidates of 1 to 10 latent components.

Figure 3 shows the graph of the selected number of latent components in SBM. The number selected by AIC increased most rapidly, however, it over-fitted the data for large sample sizes. DNML was comparable to a-NML but slightly slower than AIC. The selected numbers of components by DNML increased a bit more slowly than AIC but converged to K_{true}.

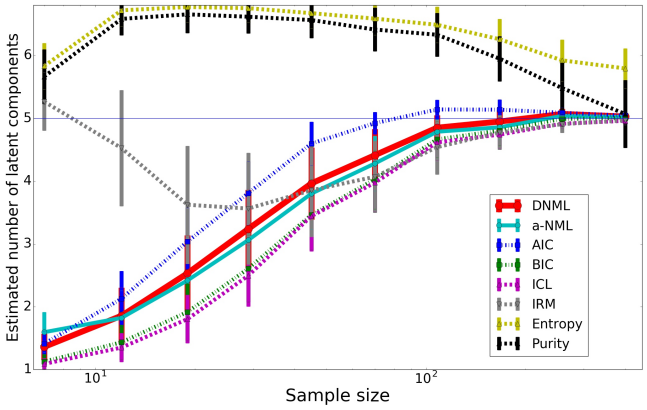

Figure 3: SBM: Estimated number K vs sample size n with $K_{true} = 5$

Besides, selection by BIC and ICL converged to the true model more slowly than the above three criteria. IRM showed unstable results for small n but converged to the true model for large sample sizes. Purity and Entropy failed to select the true model for all n.

As a result, DNML outperformed entropy and purity as a model selection criterion. We also found its advantages over BIC, ICL and IRM for small sample sizes. Meanwhile, DNML was comparable to AIC and a-NML.

4.4 Experiment 4: Model Selection: LDA, Noise-free Case

We evaluate the model selection performance of DNML for LDA. The integral in a-NML (4) for LDA cannot be calculated analytically, hence further approximate it using Monte Carlo simulation, where one iteration requires $O(DK)$ time.

As methods for comparison, in addition to AIC and BIC, we employed approximated marginal likelihoods via the Laplace approximation (Laplace) and via the evidence lower bound (ELBO), the Hierarchical Dirichlet processes (HDP) and five-fold cross validation on hold-out test dataset (CV).

For the latent variable completion in AIC and BIC, latent variables z^n are sampled from posterior distributions calculated by the variational Bayes algorithm with optimization of hyper-parameters. Laplace is a method proposed by Taddy [21]. It approximates the marginal likelihood via the Laplace method. We used the R package `maptpx` .[2] HDP [22] is a nonparametric Bayesian method which is accompanied with selection for K during inference. as in ICL. We uses the C++ source code provided by the Blei Lab [3] and utilized its default setting.

The hyper-parameters in LDA are α, β and V where $\theta_d \sim \mathrm{Dir}(\alpha)$,$\phi_k \sim \mathrm{Dir}(\beta)$ and V is the number of unique words. We generated eight datasets for each combination of hyper-parameters from 1080 candidates, which are described in [24]. The true number of latent components was set to 5 and the best model was selected from candidates with $(1, 2, 3, 4, 5, 6, 7, 8, 9, 10)$ latent components.

Table 2 shows the estimated number of topics via HDP. We observe that HDP did not converge. This is a similar finding to [12], which reported that HDP tends to select larger K as sample size increases.

Table 2: Estimated numbers of topics via HDP

Sample size	5	13	35	92	151	400
Average topics	122.1	67.6	20.5	10.6	10.5	11.6

Figure 4 shows the result except HDP. In this figure, AIC converged to the true model fastest and was able to select the true number of components even for small sample sizes. BIC and a-NML were more slower than DNML. Laplace failed to select the true number, because it is not adapted due to the irregularity of the model. ELBO and CV failed to converge to the true model. As a result, the proposed DNML outperformed HDP, Laplace, ELBO and CV. We also found its advantages over BIC and a-NML for small sample sizes. Meanwhile, DNML was comparable to AIC.

We also evaluated all the methods in terms of *perplexity*, which is defined as the exponential of the negative average

[2] https://cran.r-project.org/web/packages/maptpx/index.html, accessed Feb. 2, 2017.
[3] https://github.com/blei-lab/hdp, accessed Feb. 2, 2017.

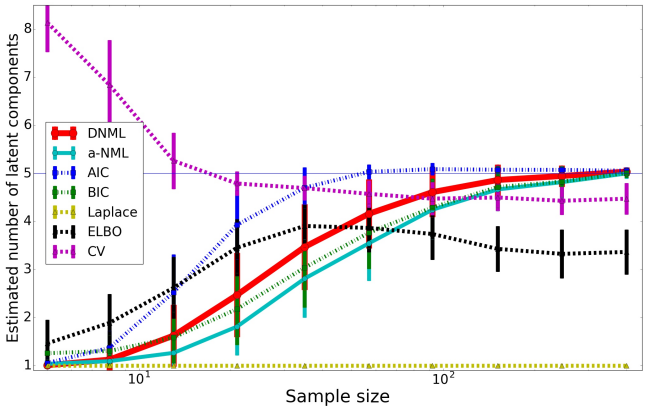

Figure 4: LDA: Estimated number K vs sample size n with $K_{true} = 5$ (HDP eliminated)

log-likelihood of a held-out test set. Since the values of perplexities depend on data distributions, we standardized them so that the best score was set to 1 by dividing all the values with the best score. Figure 5 shows the graph of the adjusted text perplexity. AIC converged fastest to the best score 1, and DNML was the second fastest. This is similar to the results on the estimation of topic numbers.

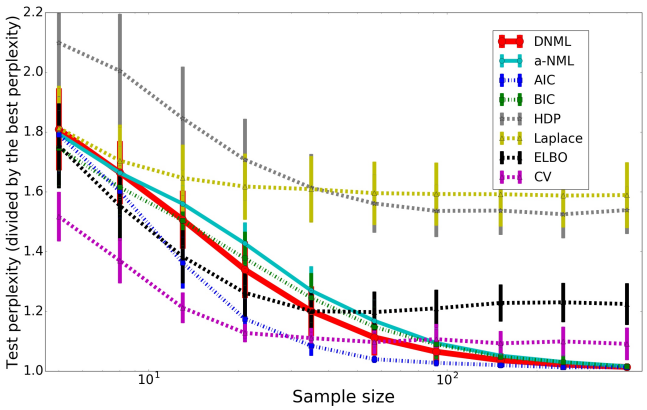

Figure 5: Test perplexities vs sample size n for the standard LDA

4.5 Experiment 5: Model Selection: LDA, Noisy Case

Next we evaluate the model selection performance of DNML for LDA with noisy observations. We added a tiny number of documents as noise as follows. We generated 8 synthetic datasets for each combination of hyper-parameters from the following candidates: $\alpha \in \{0.05, 0.1, 0.2\}$, $\beta \in \{0.1, 0.2\}$, $V \in \{300, 500\}$ and $D_{noise} \in \{0, 1, 2, 5, 10\}$, where D_{noise} was the number of documents with noise which were generated according to a five-topic model. Also, documents without noise were generated according to a five-topic model different from the one for the noise. We set $D = 800$ as the number of such documents for each simulation dataset.

In this experiment, although the true number K_{true} was no longer five, the main five topics occupied most of the proportion of the data and the proportion of noise documents was too small to be treated as a topic.

Figure 6 shows how the number of noise documents affects the absolute gap between estimated number and K_{true}. We observe that AIC was so sensitive to additional documents that it tended to select a large K even for a small number of documents. On the other hand, DNML a-NML and BIC performed robustly. Hence, DNML turns out to be robust against noise, while AIC tends to overfit noisy data.

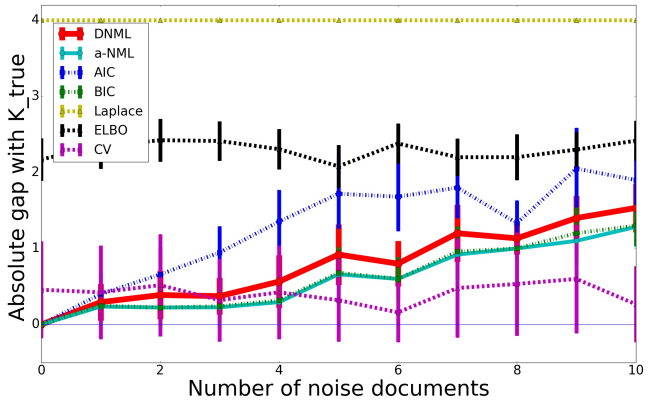

Figure 6: LDA: Absolute gap with K_{true} vs noise documents D_{noise}

5 EXPERIMENTS: REAL DATA

In this section, we evaluate the validity of DNML using two benchmark datasets: 20 Newsgroups and Reuters 21578. We used single-labeled documents and treated the labels as the underlying topics. Since this label was assigned to a document, all words in the document shared this label.

5.1 Experiment 6: 20 Newsgroups

20 Newsgroups is a collection of approximately 20,000 news documents, categorized into six major clusters and 20 sub-newsgroups, each having a label. We generated five datasets, each of which had 2,3,4,5,6 labels chosen from the original six major clusters, respectively.

We employed these datasets to evaluate how well DNML was able to estimate the number of topics, without using the label information. We implemented each method eight times, and adopted the mode of the selected numbers of topics for the eight trials.

Table 3 shows the numbers of topics estimated by DNML, a-NML, AIC, BIC, Laplace, ELBO, CV and HDP. DNML successfully selected the true number of topics for the datasets with 2, 3, 4 topics, while it selected $K_{true} + 1$ topics for the datasets with 5, 6 topics.

AIC selected larger numbers of topics than the true ones. This result is consistent with the observation for the noisy case (Experiment 5). This is because the data set contained noisy observations. a-NML and BIC performed well for most cases except the six-topic datasets. Laplace, ELBO and CV failed to select the true numbers of topics for most cases. HDP selected too many topics for all of the datasets.

Table 3: 20 Newsgroups: Selected number of topics

Method	2 topics	3 topics	4 topics	5 topics	6 topics
DNML	**2**	**3**	**4**	**6**	**7**
a-NML	**2**	**3**	**4**	**6**	3
AIC	7	4	9	7	**7**
BIC	**2**	**3**	**4**	**6**	3
Laplace	8	4	8	9	5
ELBO	3	4	**4**	4	3
CV	9	1	3	7	1
HDP	80	98	94	92	98

We further examined contents of topics selected by DNML. Table 4 shows the top words (the words that appeared most frequently) for inferred topics in the 5-topic dataset. The first topic mostly consisted of stop words. This observation is consistent with finding of [23]. For the 6-topic dataset, we also observe an additional topic for stopwords. Thus in both datasets, each of all the topics other than the first one corresponded to one of the true labels. For other datasets we confirmed that there was almost one to one correspondence between the selected topics and the true labels. This result indicates that DNML was able to successfully identify all of the true topics for these real datasets.

Table 4: Top words in inferred topic distributions selected by DNML for the 5-topic dataset

Topic	Top words	Label
1	**happen, got, someon, mayb, tell**	**stopword**
2	space, orbit, launch, nasa, earth	space
3	gun, law, state, weapon, firearm	guns
4	christian, church, jesu, sin, christ, word	christian
5	game, hit, team, player, run, pitch	baseball
6	imag, file, edu, jpeg, program, format	graphics

5.2 Experiment 7: Reuters 21578

For Reuters 21578, we used prepared data of [3] focusing on documents with a single topic. We also filtered labels with less than 50 documents. Then, we had 4 most frequent classes and generated 2, 3 and 4-topic datasets from it.

Table 5 shows the numbers of topics estimated by all the methods. DNML successfully selected the true number for all of the 3 datasets. AIC selected more topics than the true labels as in the artificial data analysis with noisy documents and 20 Newsgroups. a-NML and BIC selected less topics than the true labels due to the relatively small sample sizes in Reuters 21578. Laplace, ELBO, CV and HDP showed similar performance as in 20 Newsgroups. Following the same approach in 20 Newsgroups, we also confirmed that for each inferred topic distribution of the model selected by DNML, there was a unique correspondent major label. It indicates that DNML successfully selected the right model.

Table 5: Reuters 21578: Selected number of topics

Method	2 topics	3 topics	4 topics
DNML	**2**	**3**	**4**
a-NML	**2**	**3**	3
AIC	6	4	5
BIC	**2**	**3**	3
Laplace	8	4	8
ELBO	3	**3**	3
CV	**2**	4	1
HDP	102	112	109

6 DISCUSSION

We see from Sections 4 and 5 that DNML showed better or at least comparable performance in comparison with existing model selection methods. For most sample sizes, DNML performed much better than CV (Exp. 2,4,5,6,7), Entropy, Purity (Exp. 2, 3), IRM (Exp. 3), Laplace (Exp. 4,5,6,7), and HDP (Exp. 4,6,7). (Exp.= Experiment). ICL and ELBO performed better than DNML for small samples, but they were more unstable and worse than DNML as sample size increased (Exp. 3,4). ELBO was outperformed by DNML in real data experiments (Exp 6,7). Compared with BIC, DNML had better performance for small sample sizes and converged to the true model more rapidly in synthetic data (Exp. 3,4) and outperformed in real data (Exp. 6,7).

Compared with AIC, DNML were comparable or slightly worse for noiseless domains (Exp. 3,4). However, we see from Exp. 5 that AIC was so sensitive to noise, while DNML was robust against noise. This property of DNML is advantageous in real data analysis, because real data are likely to contain noisy observations (Exp. 6,7). Actually, the number of components selected by AIC increased more rapidly than DNML as sample size increased, then led to the poorer performance than DNML. This is because the penalty term in AIC was so small that it tended to overfit data.

Compared with a-NML, DNML converged faster to the true model in simulation data (Exp. 2,3,4,5) and had better performance in real data (Exp. 6,7). The superiority of DNML over a-NML is outstanding for small data, because a-NML is derived using asymptotic theory, while DNML is exactly evaluated. Both of a-NML and DNML can be considered as approximations of the exact NML. The approximation accuracy of DNML to the exact NML is higher than a-NML (Exp.1). Further, a-NML and DNML are both computable in linear time with respect to sample size. It implies that DNML is more preferable to a-NML for model selection of hierarchical latent variable models.

7 CONCLUSION

This paper presented the new DNML codelength as a model selection criterion for hierarchical latent variable models. To calculate this codelength, we encode latent variables and observed data separately. We have shown that this decomposition overcomes the computational difficulty of the NML codelength. Furthermore, DNML approximates NML well from both theoretical and empirical perspectives. We derived closed-form expressions for DNML on several models. We empirically demonstrated that DNML worked significantly better than conventional methods using synthetic data and real data. Our criterion especially performed well when sample size was small or data contained noises. Future studies include applications of DNML into a wider class including neural networks, etc.

ACKNOWLEDGMENTS

This research was supported by JST CREST NO. JPMJCR1304.

REFERENCES

[1] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. 2008. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research* 9 (2008), 1981–2014.

[2] H. Akaike. 1974. A new look at the statistical model identification. *IEEE Trans. on Automatic Control* 19, 6 (1974), 716–723.

[3] C. C. Ana. 2007. Improving Methods for Single-label Text Categorization. PhD Thesis, Instituto Superior Tecnico, Universidade Tecnica de Lisboa. (2007).

[4] D. M. Blei and M. I. Jordan. 2003. Modeling annotated data. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. ACM, 127–134.

[5] D. M Blei, A. Y. Ng, and M. I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3 (2003), 993–1022.

[6] J. J. Daudin, F. Picard, and S. Robin. 2008. A mixture model for random graphs. *Statistics and Computing* 18, 2 (2008), 173–183.

[7] S. Hirai and K. Yamanishi. 2013. Efficient computation of normalized maximum likelihood codes for Gaussian mixture models with its applications to clustering. *IEEE Transactions on Information Theory* 59, 11 (2013), 7718–7727.

[8] Y. Ito, S. Oeda, and K. Yamanishi. 2016. Rank Selection for Non-negative Matrix Factorization with Normalized Maximum Likelihood Coding. In *Proceedings of the 2016 SIAM International Conference on Data Mining*. SIAM, 720–728.

[9] C. Kemp, J. B. Tenenbaum, T. L. Griffiths, T. Yamada, and N. Ueda. 2006. Learning systems of concepts with an infinite relational model. In *AAAI*, Vol. 3. 5.

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

[11] P. Kontkanen, P. Myllym{"a}ki, W. Buntine, J. Rissanen, and H. Tirri. 2005. An MDL Framework for Data Clustering. In *Advances in Minimum Description Length: Theory and Applications*. MIT Press, 323.

[12] J. W. Miller and M. T. Harrison. 2013. A simple example of Dirichlet process mixture inconsistency for the number of components. In *Advances in neural information processing systems*. 199–206.

[13] J. Rissanen. 1998. *Stochastic complexity in statistical inquiry*. Vol. 15. World Scientific.

[14] J. Rissanen. 2012. *Optimal Estimation of Parameters:*. Cambridge University Press.

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

[16] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P.S Smyth. 2004. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*. AUAI Press, 487–494.

[17] Y. Sakai and K. Yamanishi. 2013. An NML-based model selection criterion for general relational data modeling. In *2013 IEEE International Conference on Big Data*. IEEE, 421–429.

[18] G. Schwarz. 1978. Estimating the dimension of a model. *The Annals of Statistics* 6, 2 (1978), 461–464.

[19] Y. M. Shtar'kov. 1987. Universal sequential coding of single messages. *Problemy Peredachi Informatsii* 23, 3 (1987), 3–17.

[20] T. A. B. Snijders and K. Nowicki. 1997. Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *Journal of Classification* 14, 1 (1997), 75–100.

[21] M. Taddy. 2012. On Estimation and Selection for Topic Models.. In *AISTATS*. 1184–1193.

[22] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei. 2012. Hierarchical dirichlet processes. *J. Amer. Statist. Assoc.* 101, 476 (2012), 1566–1581.

[23] H. M. Wallach, D. M. Mimno, and A. McCallum. 2009. Rethinking LDA: Why priors matter. In *Advances in neural information processing systems*. 1973–1981.

[24] T. Wu, S. Sugawara, and K. Yamanishi. 2017. Supplemental materials for "Decomposed Normalized Maximum Likelihood Codelength Criterion for Selecting Hierarchical Latent Variable Models". (2017). https://sites.google.com/site/shinyasugawara2012/wu17_supp2.pdf.

APPENDIX

Proof sketch for Theorem 2.1

For the codelength of $\log P(x^n, z^n; \hat{\theta}(x^n, z^n))$, this term can be decomposed into the sum of $\log P(x^n|z^n; \hat{\theta}(x^n, z^n), M)$ and $\log P(z^n; \hat{\theta}(z^n), M)$ in hierarchical latent variable models. Thus this part is same in both DNML and NML. We denote the negative value of this term as $\mathcal{L}_{data}$.

The logarithm of the probabilistic distribution of a finite mixture model can be written as $\log P(x, z) = \sum_k z_k \log \pi_k + z_k \log P(x|z_k = 1)$. Its Fisher information matrix I_{total} is derived as a block-diagonal matrix whose diagonal components are $I_{\mathrm{MN}}, \pi_1^{K^1_{base}} I^1_{base}, \cdots, \pi_K^{K^K_{base}} I^K_{base}$, where I_{MN} and I^k_{base} are the Fisher information matrices for the multinomial distribution and for the kth base distribution.

Using (4), we can compute the NML codelength as

$$
\begin{aligned}
&\mathcal{L}_{\mathrm{NML}}(x^n, z^n; M) \\
=&\mathcal{L}_{data} + \frac{K_{total}}{2} \log \frac{n}{2\pi} + \log \int \sqrt{|I_{total}|} d\theta + o(1) \\
=&\mathcal{L}_{data} + \sum_k \left\{ \log \int \sqrt{|I^k_{base}|} d\theta + \log \Gamma\left(\frac{K^k_{base} + 1}{2} \right) \right\} \\
&+ \frac{K_{total}}{2} \log \frac{n}{2\pi} - \log \Gamma\left(\frac{\sum_k K^k_{base} + K}{2} \right) + o(1).
\end{aligned}
$$

For the DNML codelength,

$$
\begin{aligned}
&\mathcal{L}_{\mathrm{DNML}}(x^n, z^n; M) \\
=&\mathcal{L}_{data} + \frac{K_{total}}{2} \log \frac{n}{2\pi} + \log \frac{\pi^{\frac{K}{2}}}{\Gamma(\frac{K}{2})} \\
&+ \sum_k \left\{ \frac{K^k_{base}}{2} \log c_k + \log \int \sqrt{|I^k_{base}|} d\theta \right\} + o(1).
\end{aligned}
$$

Subtracting $\mathcal{L}_{\mathrm{DNML}}(x^n, z^n; M)$ from $\mathcal{L}_{\mathrm{NML}}(x^n, z^n; M)$, we obtain the main equation for the theorem. □