Exact learning augmented naive Bayes classifier

Shouta Sugahara SUGAHARA@AI.LAB.UEC.AC.JP
Masaki Uto UTO@AI.LAB.UEC.AC.JP
Maomi Ueno UENO@AI.IS.UEC.AC.JP
Department of Computer and Network Engineering
The University of Electro-Communications
Tokyo, Japan

Abstract

For classification problems, Bayesian networks are often used to infer a class variable when given feature variables. Earlier reports have described that classification accuracies of Bayesian networks achieved by maximizing the marginal likelihood (ML) were lower than those achieved by maximizing the conditional log likelihood (CLL) of a class variable given the feature variables. However, the reports stated no reason why CLL outperformed ML. Differences between the two scores' performances in those earlier studies might depend on their respective learning algorithms: they were approximate learning algorithms, not exact ones. The present study compared the classification performances of Bayesian networks with exact learning using ML and those with approximate learning using CLL. Results demonstrate that the performance of Bayesian networks achieved by maximizing ML is not necessarily worse than that achieved by maximizing CLL. However, the results also show that classification accuracies with exact learning by ML are much worse than those by other methods when the class variable has numerous parents and few children. To resolve this difficulty, this study proposed exact learning augmented naive Bayes (ANB) using Markov blanket feature selection. Some comparison experiments demonstrated that the proposed method outperforms the other methods.

Keywords: Bayesian networks; classification; structure learning; augmented naive Bayes

1. Introduction

Classification, which is the inference of labels assigning the objective class from related features data, plays an important role in real-world problems.

The naive Bayes classifier, in which the feature variables are conditionally independent given a class variable, is a popular classifier (Minsky, 1961). Initially, the naive Bayes was not expected to provide highly accurate classification because actual data generation models are more complex. Therefore, the general Bayesian network (GBN) was expected to outperform the naive Bayes, but Friedman et al. (1997) demonstrated that the naive Bayes sometimes outperformed GBN using greedy search to find the smallest Minimum Description Length (MDL) score, which was originally intended to approximate marginal likelihood (ML). They explained the reason for this intention by decomposing the MDL into the log likelihood (LL) term, which reflects model fitting to training data and the penalty term which reflects the model complexity. Moreover, they decomposed the term LL into a conditional log likelihood (CLL) of the class variable, which is related directly to the classification and a joint log likelihood of the feature variables, which is not related directly to the classification.

Consequently, they claimed that conditional MDL (CMDL) score, which is a modified MDL replacing the term LL by the term CLL, should be minimized to achieve a Bayesian network with

highly accurate classification. However, unfortunately, the CLL has no closed-form equation to estimate the optimal parameters. This fact implies that some optimization algorithms must be employed, such as gradient descent over the space of parameters (e.g., Extended Logistic Regression algorithm (Greiner and Zhou, 2002)). When learning the network structure, this search must be repeated for each structure candidate, which renders the method computationally expensive.

As a simple solution to the problem, Friedman et al. (1997) proposed the augmented naive Bayes classifier (ANB) in which the class variable directly links to all feature variables and for which links among feature variables are allowed. Actually, ANB ensures that all feature variables can contribute to the classification. Later, restricted ANBs of various kinds were proposed, such as Tree-augmented naive Bayes (TAN) (Friedman et al., 1997) and Forest-augmented naive Bayes (FAN) (Lucas, 2002).

Because maximizing CLL is difficult, various approximation methods to maximize CLL have been proposed. Carvalho et al. (2013) proposed the approximate conditional log likelihood (aCLL) score, which is decomposable and computationally efficient. Furthermore, Grossman and Domingos (2004) proposed a learning structure method using a greedy hill-climbing algorithm (Heckerman et al., 1995) to maximize CLL approximately. They experimentally demonstrated that the proposed methods outperformed methods using the greedy hill-climbing algorithm to minimize MDL.

However, they stated no reason why CLL outperformed ML. Differences between the two scores' performances in those earlier studies might depend on their respective learning algorithms: they were approximate learning algorithms, not exact ones.

Recently, many algorithms of exact learning GBN to maximize ML were proposed (Silander and Myllymäki, 2006; Yuan and Malone, 2013; Cussens, 2012).

First, this study compares the classification performances of the Bayesian networks exactly learned by ML and those approximately learned by CLL. Results show that maximizing ML does necessarily not provide worse classification accuracies than maximizing CLL does. However, the results also show that classification accuracies with exact learning by ML are much worse than those by other methods when the class variable has numerous parents and few children in the exactly learned networks. When a class variable has numerous parents, the estimation of conditional probability parameters of the class variable becomes unstable because the number of patterns of the parents' values becomes large. Then the sample size for learning the parameters becomes sparse. However, the conditional probability parameters of the children given the class variable can be estimated stably as sufficiently reflecting the data because the number of parameters increases linearly as the number of children increases. Therefore, a small number of children of the class variable might be unable to reflect the feature data for classification when the sample size is insufficiently large.

To resolve the difficulty, this study proposes an exact learning ANB by maximizing ML over the class variable's *Markov blanket* in the exactly learned GBN, which is a set of relevant variables affecting the classification in the GBN structure. The proposed algorithm uses dynamic programming (DP). This employs the Bayesian Dirichlet equivalent uniform (BDeu) score: one of the most popular learning scores (Heckerman et al., 1995; Buntine, 1991). Some comparison experiments conducted with other methods show that the proposed method outperforms the other methods.

2. Bayes classifiers

2.1 Bayesian network

Let $\mathbf{X} = \{X_0, X_1, \cdots, X_n\}$ be a set of $n+1$ discrete variables; $X_i, (i = 0, \cdots, n)$ can take values in the set of states $\{1, \cdots, r_i\}$. We write $X_i = k$ when we observe that an X_i is state k. According to the Bayesian network structure S, the joint probabilities distribution is given as

$$P(X_0, X_1, \cdots, X_n) = \prod_{i=0}^{n} P(X_i \mid \Pi_i, S), \tag{1}$$

where Π_i is the parent variable set of X_i. Letting θ_{ijk} be a conditional probability parameter of $X_i = k$ when the j-th instance of the parents of X_i is observed (We write $\Pi_i = j$), we define $\Theta = \{\theta_{ijk}\}$ $(i = 0, \cdots, n; j = 1, \cdots, q_i; k = 1, \cdots, r_i)$. A Bayesian network is a pair $B = (S, \Theta)$. Buntine (1991) assumed the Dirichlet prior and used an expected a posteriori (EAP) estimator $\hat{\theta}_{ijk}$:

$$\hat{\theta}_{ijk} = \frac{\alpha_{ijk} + N_{ijk}}{\alpha_{ij} + N_{ij}}. \tag{2}$$

In that equation, N_{ijk} represents the number of samples of $X_i = k$ when $\Pi_i = j$, $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$. Also, α_{ijk} denotes the hyperparameters of the Dirichlet prior distributions (α_{ijk} is a pseudo-sample corresponding to N_{ijk}); $\alpha_{ij} = \sum_{k=1}^{r_i} \alpha_{ijk}$.

The first learning task of the Bayesian network is to seek a structure S optimizing a given score. The most popular marginal likelihood (ML) score of Bayesian network (using a Dirichlet prior over model parameters) finds the maximum a posteriori (MAP) structure when we assume a uniform prior over structures, as described by Buntine (1991) and Heckerman et al. (1995). In addition, the Dirichlet prior is known as a distribution that ensures likelihood equivalence. This score is known as *Bayesian Dirichlet equivalence (BDe)* (Heckerman et al., 1995). Given no prior knowledge, *the Bayesian Dirichlet equivalence uniform (BDeu)*, as proposed earlier by Buntine (1991), is often used. Let $D = \{\mathbf{x}^1, \cdots, \mathbf{x}^d, \cdots, \mathbf{x}^N\}$ be training dataset and let each $\mathbf{x}^d$ be a tuple of the form $\langle x_0^d, x_1^d, \cdots, x_n^d \rangle$. For the analyses presented in this paper, we assume no missing data throughout. The BDeu score is represented as

$$P(D \mid S) = \prod_{i=0}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(\frac{\alpha}{q_i})}{\Gamma(\frac{\alpha}{q_i} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\frac{\alpha}{r_i q_i} + N_{ijk})}{\Gamma(\frac{\alpha}{r_i q_i})}, \tag{3}$$

where α is a hyperparameter.

The Minimum Description Length (MDL), which approximates ML, is also often used, as presented below.

$$MDL(B \mid D) = \frac{\log N}{2}|\Theta| - \sum_{d=1}^{N} \log P(x_0^d, x_1^d, \cdots, x_n^d \mid B), \tag{4}$$

$|\Theta|$ represents the number of parameters. Consequently, the first term is the *penalty term*, which signifies the model complexity. The second term, log likelihood (LL), is the *fitting term*, which represents model fitting to the training data.

2.2 Bayes classifiers

A Bayes classifier can be interpreted as a Bayesian network for which X_0 is the class variable and for which $X_1, \cdots, X_n$ are feature variables. Given an instance $\langle x_1, \cdots, x_n \rangle$ for feature variables $X_1, \ldots, X_n$, the Bayes classifier B infers the class c by maximizing the posterior probability as

$$\begin{aligned} \hat{c} &= \underset{c \in \{1,\cdots,r_0\}}{\arg\max}\, P(c \mid x_1, \cdots, x_n, B) = \underset{c \in \{1,\cdots,r_0\}}{\arg\max} \prod_{i=0}^{n} \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} (\theta_{ijk})^{1_{ijk}} \\ &= \underset{c \in \{1,\cdots,r_0\}}{\arg\max} \prod_{j=1}^{q_0} \prod_{k=1}^{r_0} (\theta_{0jk})^{1_{0jk}} \times \prod_{i:X_i \in \mathbf{C}} \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} (\theta_{ijk})^{1_{ijk}}, \end{aligned} \tag{5}$$

where $1_{ijk} = 1$ if $X_i = k$ and $\Pi_i = j$ in case $\langle x_1, \cdots, x_n \rangle$ and $1_{ijk} = 0$ otherwise. Furthermore, $\mathbf{C}$ is a set of children of the class variable X_0. From Equation 5, we can infer class c given only the values of the X_0's parents, the X_0's children, and the parents of the X_0's children, which are called a *Markov blanket* of X_0.

However, Friedman et al.(1997) reported that the Bayes classifier minimizing MDL can not optimize the classification performance. They proposed the sole use of the conditional log likelihood (CLL) of the class variable given the feature variables instead of the log likelihood for learning Bayes classifier structures. Consequently, they proposed conditional MDL (CMDL), which is a modified MDL replacing LL by CLL, as shown below.

$$CMDL(B \mid D) = \frac{\log N}{2}|\Theta| - \sum_{d=1}^{N} \log P(x_0^d \mid x_1^d, \cdots, x_n^d, B) \tag{6}$$

Unfortunately, no closed-form formula exists for optimal parameter estimates to maximize CLL. Therefore, for each structure candidate, learning the network structure minimizing CMDL requires some search methods such as gradient descent over the space of parameters. For that reason, exact learning network structures by minimizing CMDL is computationally infeasible.

As a simple means of solving the problem, Friedman et al. (1997) proposed the augmented naive Bayes classifier (ANB), which ensures an edge from the class variable to each feature variable and which allows edges among feature variables. Furthermore, they proposed tree-augmented naive Bayes (TAN), in which the class variable has no parents and in which each feature variable has the class variable and at most one other feature variable as a parent variable.

On the other hand, various approximate methods to maximize CLL have been proposed. Carvalho et al. (2013) proposed an approximate conditional log likelihood (aCLL) score, which is decomposable and computationally efficient. Let S_{ANB} be an ANB structure of B. Then we define $\Pi_i^* = \Pi_i \setminus \{X_0\}$ based on S_{ANB}. In addition, let N_{ijck} be the number of samples of $X_i = k$ when $X_0 = c$ and $\Pi_i^* = j$ $(i = 1, \cdots, n; j = 1, \cdots, q_i^*; c = 1, \cdots, r_0; k = 1, \cdots, r_i)$, and let $N' > 0$ be the number of pseudo-counts. Under several assumptions, aCLL can be represented as

$$aCLL(S_{ANB} \mid D) \propto \sum_{i=1}^{n} \sum_{j=1}^{q_i^*} \sum_{k=1}^{r_i} \sum_{c=1}^{r_0} \left(N_{ijck} + \beta \sum_{c'=1}^{r_0} N_{ijc'k} \right) \log \frac{N_{ij+ck}}{N_{ij+c}}, \tag{7}$$

where

$$N_{ij+ck} = \begin{cases} N_{ijck} + \beta \sum_{c'=1}^{r_0} N_{ijc'k} & \textit{if } N_{ijck} + \beta \sum_{c'=1}^{r_0} N_{ijc'k} \geq N' \\ N' & \textit{otherwise}, \end{cases}$$

$$N_{ij+c} = \sum_{k=1}^{r_i} N_{ij+ck}.$$

A value of β is found using a Monte-Carlo method to approximate CLL. When the value of β is optimal, aCLL is a minimum-variance unbiased approximation of CLL.

Moreover, Grossman and Domingos (2004) proposed a learning structure method using a greedy hill-climbing algorithm (Heckerman et al., 1995) by maximizing CLL while choosing parameters by maximizing LL.

These reports described that the classifier maximizing the approximated CLL provides better performance than that maximizing the approximated ML.

However, they specified no reason why CLL outperformed ML. Differences of performance between MDL and CLL in earlier studies might depend on the employed learning algorithms because they used not exact but approximate learning algorithms.

3. Classification accuracy of exact learning GBN

This section presents experiments comparing the classification accuracy of the exactly learned GBN by maximizing ML with that of the approximately learned network structure methods by maximizing CLL. Although BDeu has rarely been used for Bayesian classifiers, de Campos et al. (2014) proposed an extended TAN classifier by maximizing the BDeu score. The results demonstrated the effectiveness of the proposed method. This study also employs BDeu for learning a structure. Although determination of hyperparameter α of BDeu is difficult (Silander et al., 2007; Steck, 2008; Ueno, 2008; Suzuki, 2017), we use $\alpha = 1.0$, which allows the data to reflect the estimated parameters to the greatest degree possible (Ueno, 2010, 2011).

The experiment compares the classification accuracies of the following six methods.

- *GBN-BDeu*: Exact learning GBN method by maximizing BDeu.
- *Naive Bayes*
- *GBN-CMDL* (Grossman and Domingos, 2004): Greedy learning GBN method using the hill-climbing by minimizing CMDL.
- *BNC2P* (Grossman and Domingos, 2004): Greedy learning method with at most two parents per variable using the hill-climbing by maximizing CLL (choosing parameters by maximizing LL).
- *TAN-aCLL* (Carvalho et al., 2013): Exact learning TAN method by maximizing aCLL.
- *gGBN-BDeu*: Greedy learning GBN method using the hill-climbing by maximizing BDeu.

We used EAP estimators as conditional probability parameters of the respective classifiers. Hyperparameters α_{ijk} of EAP were determined as $1/(r_i q_i)$.

This experiment used 43 classification benchmark datasets from the UCI repository. Continuous variables were discretized into two bins using the median value as cut-off, as in (de Campos et al., 2014). In addition, data with missing values were removed from the datasets.

Table 1 presents the results. In Table 1, the values in bold represent the best accuracies for each dataset. Here, the classification accuracy indicates the average percentage correct of classifications from ten-fold cross validation. Moreover, to investigate the relation between the classification accuracy and the GBN structure, Table 2 presents details of the achieved structures using *GBN-BDeu*.

No.	Dataset	Variables	Sample size	Classes	Naive-Bayes	GBN-CMDL	BNC2P	TAN-aCLL	gGBN-BDeu	GBN-BDeu	ANB-BDeu
1	Balance Scale	5	625	3	**0.9152**	0.3333	0.8560	0.8656	**0.9152**	**0.9152**	**0.9152**
2	banknote authentication	5	1372	2	0.8433	**0.8819**	0.8797	0.8761	**0.8819**	0.8812	0.8812
3	Hayes-Roth	5	132	3	**0.8182**	0.6136	0.6894	0.6742	0.7525	0.6136	**0.8182**
4	iris	5	150	3	0.7133	0.7800	0.8200	0.8200	0.8133	**0.8267**	0.8200
5	lenses	5	24	3	0.7500	**0.8333**	0.6667	0.7083	**0.8333**	**0.8333**	0.7500
6	Car Evaluation	7	1728	4	0.8571	**0.9497**	0.9416	0.9433	0.9416	0.9416	0.9427
7	liver	7	345	2	0.6319	0.6145	0.6290	**0.6609**	0.6029	0.6087	0.6348
8	MONK's Problems	7	432	2	0.7500	**1.0000**	**1.0000**	**1.0000**	0.8449	**1.0000**	**1.0000**
9	mux6	7	64	2	0.5469	0.3750	**0.5625**	0.4688	0.4063	0.4531	0.5469
10	led7	8	3200	10	0.7294	0.7366	**0.7375**	0.7350	0.7297	0.7294	0.7294
11	HTRU2	9	17898	2	0.7031	0.7096	0.7070	0.7018	0.7188	**0.7305**	0.7188
12	Nursery	9	12960	3	0.6782	**0.7126**	0.6092	0.5862	**0.7126**	**0.7126**	0.6782
13	pima	9	768	9	0.8966	0.9086	0.9118	0.9130	0.9092	0.9112	**0.9141**
14	post	9	87	5	0.9033	0.5823	**0.9442**	0.9177	0.9291	0.9340	0.9181
15	Breast Cancer	10	277	2	**0.9751**	0.8917	0.9473	0.9488	0.7058	**0.9751**	**0.9751**
16	Breast Cancer Wisconsin	10	683	2	**0.7401**	0.6209	0.6823	0.7184	0.7094	0.7184	0.7040
17	Contraceptive Method Choice	10	1473	3	0.4671	0.4501	**0.4745**	0.4705	0.4440	0.4542	0.4650
18	glass	10	214	6	0.5561	0.5654	0.5794	0.6308	0.4626	0.5701	**0.6449**
19	shuttle-small	10	5800	6	0.9384	0.9660	0.9703	0.9583	0.9683	0.9693	**0.9716**
20	threeOf9	10	512	2	0.8164	**0.9434**	0.8691	0.8828	0.8652	0.8887	0.8730
21	Tic-Tac-Toe	10	958	2	0.6921	**0.8841**	0.7338	0.7203	0.6754	0.8340	0.8497
22	MAGIC Gamma Telescope	11	19020	2	0.7482	0.7849	0.7806	0.7631	0.7844	0.7873	**0.7874**
23	Solar Flare	11	1389	9	0.7811	0.8265	0.8315	0.8229	**0.8431**	**0.8431**	0.8229
24	heart	14	270	2	**0.8259**	0.8185	0.8037	0.8148	0.8222	**0.8259**	0.8185
25	wine	14	178	3	0.9270	**0.9438**	0.9157	0.9326	0.9045	0.9270	0.9270
26	cleve	14	296	2	**0.8412**	0.8209	0.8007	0.8378	0.7973	0.7973	0.8277
27	australian	15	690	2	0.8290	0.8312	0.8348	0.8464	0.8420	**0.8536**	0.8246
28	crx	15	653	2	0.8377	0.8346	0.8208	0.8560	**0.8622**	0.8591	0.8515
29	EEG	15	14980	2	0.5778	0.6787	0.6374	0.6125	0.6732	0.6814	**0.6864**
30	Congressional Voting Records	17	232	2	0.9095	0.9698	0.9612	0.9181	**0.9741**	0.9655	0.9483
31	zoo	17	101	5	0.9802	0.9109	0.9505	**1.0000**	0.9505	0.9307	0.9505
32	pendigits	17	10992	10	0.8032	0.9062	0.8719	0.8700	0.9253	**0.9290**	0.9279
33	letter	17	20000	26	0.4466	0.5796	0.5132	0.5093	0.5761	0.5761	**0.5935**
34	ClimateModel	19	540	2	0.9222	**0.9407**	0.9241	0.9333	0.9370	0.9000	0.8426
35	Image Segmentation	19	2310	7	0.7290	0.7918	0.7991	0.7407	0.8026	0.8156	**0.8225**
36	lymphography	19	148	4	**0.8446**	0.7939	0.7973	0.8311	0.7905	0.7500	0.7770
37	vehicle	19	846	4	0.4350	0.5910	0.5910	0.5816	0.5461	0.5768	**0.6253**
38	hepatitis	20	80	2	0.8500	0.7375	**0.8875**	0.8750	0.8500	0.5875	0.6250
39	german	21	1000	2	0.7430	0.6110	0.7340	**0.7470**	0.7140	0.7210	0.7380
40	bank	21	30488	2	0.8544	0.8618	0.8928	0.8618	0.8952	**0.8956**	0.8950
41	waveform-21	22	5000	3	0.7886	0.7862	0.7754	0.7896	0.7698	0.7846	**0.7966**
42	Mushroom	22	5644	2	0.9957	**1.0000**	**1.0000**	0.9995	**1.0000**	0.9949	**1.0000**
43	spect	23	263	2	0.7940	0.7940	0.7903	0.8090	0.7603	0.7378	**0.8240**
	average				0.7764	0.7721	0.7936	0.7943	0.7867	0.7963	**0.8061**
	p-value				0.0031	0.0414	0.0067	0.0561	0.0629	0.2263	-

Table 1: Accuracies of respective classifiers for 43 datasets

"Parents" in Table 2 presents the average maximum number of parents of the class variable from the *GBN-BDeu* based learned structures, "Children" denotes the average number of children of the class variable from the *GBN-BDeu* based learned structures. "Sparse data" denotes the average number of patterns of X_0's parents value j with null data, $N_{0j} = 0$ $(j = 1, \cdots, q_0)$ from the *GBN-BDeu* based on learned structures. "MBsize" represents the average number of the Markov blanket size from *GBN-BDeu* based on learned structures.

From Table 1, *GBN-BDeu* shows the best average accuracy among the methods explained above. This result suggests that the performances of Bayesian networks by maximizing ML are not necessarily worse than those by maximizing by CLL. However, it is noteworthy that *GBN-BDeu* provides much worse accuracies than those for the other methods do in No. 3 and No. 9 datasets. In these datasets, the learned class variables by *GBN-BDeu* have no child. Numerous parents are shown in "Parents" and "Children" of Table 2. When a class variable has numerous parents, estimation of

No.	Dataset	Variables	Classes	Sample size	Parents	Children	Sparse data	MBsize	Missing variables	Extra variables	Max parents
1	Balance Scale	5	3	625	0.4	3.6	0.0	4.0	0.0	0.0	1.0
2	banknote authentication	5	2	1372	0.0	2.0	0.0	4.0	0.0	0.0	4.0
3	Hayes–Roth	5	3	132	3.0	0.0	17.2	3.0	0.0	0.0	1.0
4	iris	5	3	150	1.8	1.2	0.0	3.0	0.0	0.0	2.0
5	lenses	5	3	24	1.1	1.0	0.0	2.0	0.0	0.1	1.1
6	Car Evaluation	7	4	1728	2.0	3.0	0.0	5.0	0.0	0.0	2.0
7	liver	7	2	345	0.0	1.9	0.0	3.4	1.6	0.0	2.0
8	MONK's Problems	7	2	432	3.0	0.0	0.0	3.0	0.0	0.0	3.0
9	mux6	7	2	64	5.8	0.0	5.2	5.8	0.2	0.0	1.0
10	led7	8	10	3200	0.9	6.1	0.0	7.0	0.0	0.0	1.0
11	HTRU2	9	2	17898	1.8	4.2	0.0	7.0	0.0	0.0	3.0
12	Nursery	9	5	12960	4.0	3.0	0.0	7.0	0.0	0.0	3.0
13	pima	9	2	768	1.4	1.7	0.0	4.0	0.0	0.2	2.0
14	post	9	3	87	0.0	0.0	0.0	0.0	0.0	0.0	0.0
15	Breast Cancer	10	2	277	0.9	8.0	0.0	8.0	0.0	0.9	1.0
16	Breast Cancer Wisconsin	10	2	683	0.7	0.3	0.0	1.0	0.0	0.0	1.0
17	Contraceptive Method Choice	10	3	1473	0.7	0.8	0.0	1.5	0.5	0.2	1.2
18	glass	10	6	214	0.6	3.1	0.0	4.2	0.8	0.1	1.6
19	shuttle-small	10	6	5800	2.0	4.0	0.0	7.0	0.0	0.0	5.0
20	threeOf9	10	2	512	5.0	2.1	0.0	7.6	1.4	0.0	2.4
21	Tic-Tac-Toe	10	2	958	1.2	2.2	0.0	5.1	0.9	0.2	3.0
22	MAGIC Gamma Telescope	11	2	19020	0.0	6.1	0.0	8.0	0.0	0.0	4.0
23	Solar Flare	11	9	1389	0.8	0.2	0.0	0.1	0.9	0.9	1.0
24	heart	14	2	270	1.8	4.2	0.0	6.0	1.0	0.3	1.5
25	wine	14	3	178	1.7	5.3	0.0	7.0	0.0	1.1	2.1
26	cleve	14	2	296	1.8	4.5	0.0	6.3	0.7	0.3	2.0
27	Australian	15	2	690	1.4	2.8	0.0	4.2	0.8	0.3	2.2
28	crx	15	2	653	1.3	2.8	0.0	2.9	1.1	1.3	2.0
29	EEG	15	2	14980	0.4	8.2	0.0	12.8	0.2	0.0	5.0
30	Congressional Voting Records	17	2	232	1.3	2.6	0.1	5.2	1.8	1.0	2.8
31	zoo	17	5	101	4.3	1.6	20.3	6.9	3.1	0.5	3.5
32	pendigits	17	10	10992	2.6	13.4	0.1	16.0	0.0	0.0	5.6
33	letter	17	26	20000	2.9	9.1	0.0	13.0	0.0	0.0	5.0
34	ClimateModel	19	2	540	1.8	4.4	0.0	15.9	1.1	0.7	14.0
35	Image Segmentation	19	7	2310	0.7	10.4	0.0	12.7	0.3	0.5	4.1
36	lymphography	19	4	148	1.6	5.9	0.2	9.0	1.0	4.1	8.0
37	vehicle	19	4	846	1.1	5.1	0.1	9.0	2.0	1.1	3.6
38	hepatitis	20	2	80	1.3	6.1	0.4	13.1	1.9	2.9	10.7
39	German	21	2	1000	1.1	2.8	0.0	3.9	2.1	0.2	1.2
40	bank	21	2	30488	4.1	2.0	32.5	9.9	0.1	0.0	5.0
41	waveform-21	22	3	5000	3.8	10.1	0.0	13.9	0.1	0.6	4.0
42	Mushroom	22	2	5644	1.3	3.3	9.0	6.1	12.9	0.3	5.2
43	spect	23	2	263	2.0	3.4	0.0	6.4	2.6	1.3	2.5

Table 2: Statistics summary of *GBN-BDeu* and *MANB-BDeu*

the conditional probability parameters of the class variable becomes unstable because the number of patterns of the parents values becomes large. Then the samples for the learning the parameters becomes sparse as presented in "Sparse data" of Table 2. However, the conditional probability parameters of the children given the class variable can be estimated stably as sufficiently reflecting the data because the number of parameters increases linearly as the number of the children increases. Therefore, a small number of children of the class variable might be unable to reflect the feature data for classification when the sample size is insufficiently large.

This analysis suggests that exact learning GBN by maximizing BDeu so as to have a small number of parents of the class variable and numerous children of the class variable might improve the accuracies of *GBN-BDeu*. A straightforward approach to ensure this idea is an exact learning ANB structure because the class variable has no parents: all the feature variables are children. However, *Naive Bayes* and *TAN-aCLL*, in which all feature variables are children of the class variable, provide much worse accuracy than *GBN-BDeu* does when the Markov blanket size of *GBN-BDeu* is

smaller than the number of all feature variables, as presented in Table 2. This result implies that ANB includes irrelevant feature variables with the class variable. These variables are known to have lower classification accuracy often because they only introduce noise in the classification (Langley and Sage, 1994). To avoid irrelevant variables in the feature variables, this study employs feature selection by finding the Markov blanket of a class variable.

4. Exact learning ANB

This section presents the algorithm of an exact learning method of ANB by maximizing the BDeu score over Markov Blanket of the class variable. The proposed algorithm employs dynamic programming (DP) (Silander and Myllymäki, 2006). The DP algorithm seeks an optimal score from all structures including all variables, but our algorithm searches from all structures including only feature variables always having the class variable as a parent. Specifically, the procedure includes the following four steps:

1. Find the Markov blanket $\boldsymbol{M}$ from the learned optimal GBN structure over all variables using (Silander and Myllymäki, 2006).

2. Compute the local log BDeu scores for all possible $m2^{m-1}$ $(X_i, \Pi_i^* \cup \{X_0\})$ pairs over $\boldsymbol{M} \cup \{X_0\}$, where m is size of $\boldsymbol{M}$. The local score for variable X_i given parents Π_i^* and the class variable X_0 is defined as

$$
\begin{aligned}
&Score(X_i \mid \Pi_i^*, X_0) \\
&\quad = \sum_{j=1}^{q_i^*} \sum_{c=1}^{r_0} \left[\log \left(\frac{\Gamma(\frac{1}{q_i^* r_0})}{\Gamma(\frac{1}{q_i^* r_0} + \sum_{k=1}^{r_i} N_{ijck})} \right) + \sum_{k=1}^{r_i} \log \left(\frac{\Gamma(\frac{1}{r_i q_i^* r_0} + N_{ijck})}{\Gamma(\frac{1}{r_i q_i^* r_0})} \right) \right].
\end{aligned}
$$

3. For each feature variable $X_i \in \boldsymbol{M}$, find the best parent set in candidate parent set $\boldsymbol{Z} \cup \{X_0\}$ for all $\boldsymbol{Z} \subseteq \boldsymbol{M} \setminus \{X_i\}$.

4. Find the optimal ANB structure over $\boldsymbol{M}$.

During the local score calculations and the best parent searches, the class variable X_0 is always included as a parent of all the feature variables. After step 3, one can find the optimal ANB structure using the same procedures as those proposed by Silander and Myllymäki (2006).

5. Experiments

This section presents numerical experiments that were conducted to evaluate the effectiveness of the proposed method. First, we compare the classification accuracies of exact learning ANB method without Markov-blanket-based feature selection (designated as *ANB-BDeu*) with those of the other methods in Section 3. The procedure of *ANB-BDeu* is obtained by replacing $\boldsymbol{M}$ of the procedure in section 4 by all the feature variables. To confirm the significant differences of *ANB-BDeu* from the other methods, we applied Hommel's tests (Hommel, 1988), which are used as a standard in machine learning studies (Demšar, 2006). The p-values are presented at the bottom of Table 1. Results show that *ANB-BDeu* outperforms *Naive Bayes*, *GBN-CMDL*, and *BNC2P* at the $p < 0.05$ significance level. *ANB-BDeu* improves the accuracy of *GBN-BDeu* when the class variable has numerous parents and a small number of children, which is true of the No. 3, No. 9, and No. 31 datasets, as presented in Tables 1 and 2. The difference is not statistically significant. It is

No.	Dataset	Variables	Sample size	Classes	MNaive-Bayes	MGBN-CMDL	MBNC 2P	MTAN-aCLL	MgGBN-BDeu	GBN-BDeu	MANB-BDeu
1	Balance Scale	5	625	3	**0.9152**	0.3333	0.8560	0.8656	**0.9152**	**0.9152**	**0.9152**
2	banknote authentication	5	1372	2	0.8433	**0.8819**	0.8783	0.8761	0.8812	0.8812	0.8812
3	Hayes-Roth	5	132	3	**0.8333**	0.6136	0.7197	0.7879	0.7980	0.6136	**0.8333**
4	iris	5	150	3	**0.8267**	0.7800	0.8200	0.8200	0.8200	**0.8267**	**0.8267**
5	lenses	5	24	3	0.8333	0.8333	0.8333	0.8333	**0.8750**	0.8333	0.8333
6	Car Evaluation	7	1728	4	0.8559	0.9242	0.9375	0.9363	**0.9416**	**0.9416**	**0.9416**
7	liver	7	345	2	**0.6348**	**0.6348**	0.6000	0.5942	0.6000	0.6087	0.5855
8	MONK's Problems	7	432	2	0.7500	**1.0000**	**1.0000**	**1.0000**	0.8194	**1.0000**	**1.0000**
9	mux6	7	64	2	0.5469	0.3750	**0.6250**	0.4688	0.3906	0.4531	0.5469
10	led7	8	3200	10	0.7294	0.7363	**0.7375**	0.7350	0.7303	0.7294	0.7294
11	HTRU2	9	17898	2	0.7083	0.7057	0.7044	0.7070	**0.7305**	**0.7305**	0.7227
12	Nursery	9	12960	3	**0.7126**	**0.7126**	**0.7126**	**0.7126**	**0.7126**	**0.7126**	**0.7126**
13	pima	9	768	9	0.9102	0.9046	0.9076	**0.9141**	0.9083	0.9112	**0.9141**
14	post	9	87	5	0.8996	0.8775	0.9322	0.9103	0.9258	**0.9340**	0.9174
15	Breast Cancer	10	277	2	**0.9751**	0.8909	0.9663	0.9458	0.9429	**0.9751**	**0.9751**
16	Breast Cancer Wisconsin	10	683	2	**0.7184**	**0.7184**	**0.7184**	**0.7184**	**0.7184**	**0.7184**	0.7166
17	Contraceptive Method Choice	10	1473	3	0.4549	0.4542	**0.4555**	0.4535	0.4501	0.4542	0.4549
18	glass	10	214	6	**0.5841**	0.5514	0.5467	**0.5841**	0.5047	0.5701	0.5654
19	shuttle-small	10	5800	6	0.9360	0.9645	0.9666	0.9605	0.9690	**0.9693**	**0.9693**
20	threeOf9	10	512	2	0.8145	0.8750	0.8750	0.8809	0.8652	**0.8887**	0.8711
21	Tic-Tac-Toe	10	958	2	0.7182	**0.8476**	0.7244	0.7213	0.7359	0.8340	**0.8476**
22	MAGIC Gamma Telescope	11	19020	2	0.7520	0.7841	0.7807	0.7699	0.7875	0.7873	**0.7880**
23	Solar Flare	11	1389	9	**0.8431**	**0.8431**	**0.8431**	**0.8431**	**0.8431**	**0.8431**	**0.8431**
24	heart	14	270	2	0.8222	0.8185	0.8148	0.8259	0.7889	0.8259	**0.8296**
25	wine	14	178	3	**0.9607**	0.9494	0.9438	0.9494	0.9326	0.9270	0.9326
26	cleve	14	296	2	**0.8176**	**0.8176**	0.7804	0.8108	0.7905	0.7973	0.8108
27	australian	15	690	2	0.8536	**0.8580**	0.8493	0.8522	0.8507	0.8536	0.8507
28	crx	15	653	2	**0.8622**	0.8545	0.8545	**0.8622**	0.8576	0.8591	**0.8622**
29	EEG	15	14980	2	0.5774	0.6790	0.6389	0.6111	0.6670	0.6814	**0.6935**
30	Congressional Voting Records	17	232	2	0.9353	**0.9698**	0.9655	0.9397	0.9655	0.9655	0.9569
31	zoo	17	101	5	0.9406	0.9406	0.9307	0.9307	**0.9505**	0.9307	**0.9505**
32	pendigits	17	10992	10	0.8032	0.9062	0.8719	0.8700	0.9253	**0.9290**	0.9297
33	letter	17	20000	26	0.4536	**0.5796**	0.5068	0.5036	0.5636	0.5761	0.5779
34	ClimateModel	19	540	2	0.9259	**0.9407**	0.9222	0.9352	0.9370	0.9000	0.8667
35	Image Segmentation	19	2310	7	0.7662	0.7848	0.7918	0.7922	0.8022	0.8156	**0.8203**
36	lymphography	19	148	4	**0.8176**	0.7027	0.7770	0.8041	0.7770	0.7500	0.8108
37	vehicle	19	846	4	0.4634	0.5816	0.5721	0.5922	0.5437	0.5768	**0.6028**
38	hepatitis	20	80	2	**0.8750**	0.8500	0.8625	0.8500	0.8625	0.5875	0.6625
39	german	21	1000	2	0.7210	0.7250	**0.7350**	0.7230	0.7230	0.7210	0.7240
40	bank	21	30488	2	0.8680	0.8955	0.8924	0.8777	0.8954	0.8956	**0.8966**
41	waveform-21	22	5000	3	0.7852	0.7912	0.7806	0.7814	0.7626	0.7846	**0.7920**
42	Mushroom	22	5644	2	0.9970	0.9991	0.9991	0.9972	**1.0000**	0.9949	**1.0000**
43	spect	23	263	2	**0.7865**	0.7303	0.7416	0.7715	0.7715	0.7378	0.7603
	average				0.7867	0.7801	0.7993	0.7981	0.7961	0.7963	**0.8074**
	p-value				0.0089	0.0054	0.0104	0.0057	0.0188	0.0301	-

Table 3: Accuracies of respective classifiers achieved using feature selection for 43 datasets

noteworthy that the accuracies of *ANB-BDeu* are much worse than those provided by *GBN-BDeu* for the No. 5 and No. 14 datasets. Markov blanket sizes of these datasets are much smaller than the number of all feature variables, as shown in Table 2. Results show that feature selection by the Markov blanket is expected to improve the classification accuracies of the exact learning ANB method, as described in Section 3.

Next, we conduct experiments comparing the classification accuracies of the seven methods in Table 1 using Markov blanket feature selection (We apply 'M' as a prefix to each method name in Table 1). Table 3 shows the average accuracies and p-values of Hommel's tests. Results show that *MANB-BDeu* outperforms all the compared methods at the $p < 0.05$ significance level.

"Max parents" in Table 2 presents the average maximum number of parents learned by *MANB-BDeu*. A value of "Max parents" represents the complexity of the structure learned by *MANB-BDeu*. The results show that accuracies of *MNaive Bayes* are better than those of *MANB-BDeu* when the

sample size is small, such as No. 36 and No. 38 dataset. In these datasets, values of "Max parents" are large. When a variable has numerous parents, estimation of the variable parameters tends to become unstable as described in Section 3. *MNaive Bayes* can avoid this phenomenon because the maximum number of parents is constantly only one. However, *MNaive Bayes* cannot learn relations among feature variables at all. Therefore, *MNaive Bayes* shows much worse accuracies than those for the other methods when the sample size is large such as No. 8 and No. 29 datasets.

MGBN-CMDL shows much worse accuracies than those for the other methods in No. 1, No. 3, No. 9, No. 14, and No. 15 datasets because the penalty term of CMDL does not correspond to CLL. Actually, the penalty term was not derived from the conditional marginal likelihood but from the ML. *MBNC2P* and *MTAN-aCLL* avoid CLL to prefer adding extra edges with a restriction of at most two parents per variable instead of using the penalty term. However, the small upper bound of maximum number of parents tends to cause poor representational power of the structure (Ling and Zhang, 2003). As a result, accuracies of the two methods tend to be worse than those of the *MANB-BDeu* in datasets on which the value of "Max parents" is greater than two, such as the No. 29 dataset. However, like *Naive Bayes*, the two methods show better accuracies than *MANB-BDeu* does when the sample size is small such as the No. 38 dataset.

MgGBN-BDeu also shows better accuracy than *MANB-BDeu* does for small samples, although *MgGBN-BDeu* obtains worse accuracy for large samples such as the No. 29 and No. 33 datasets. The reason is that the exact learning methods estimate the network structure more precisely than the greedy learned structure for larger samples.

MANB-BDeu improves the accuracies of *GBN-BDeu* when the class variable has numerous parents and a small number of children such as the No. 3, No. 9, and No. 31 datasets, similarly to *ANB-BDeu*. The reason is that *MANB-BDeu* avoids the sparse-data problem of *GBN-BDeu*.

Finally, we compare *MANB-BDeu* and *ANB-BDeu*. The difference of the two methods is whether a Markov blanket feature selection is used or not. "Missing variables" are the average numbers of relevant variables to a class variable that are discarded by the feature selection. "Extra variables" are the average number of irrelevant variables to a class variable that are selected by the feature selection. We do not know the true relevant variable set. Therefore, we regard it as a class variable's Markov blanket of the learned structure by *GBN-BDeu* using whole training data. Results demonstrate that accuracies of *MANB-BDeu* tend to be much better than those of *ANB-BDeu* in datasets when the value of "MBsize" is small, such as No. 5 and No. 25 datasets. Consequently, the discarding numerous irrelevant variables in the features improves the classification accuracy. The values of "Extra variables" tend to be small for almost all datasets. In contrast, the accuracies of *MANB-BDeu* tend to be worse than those of *ANB-BDeu* when the value of "Missing variables" is large, such as the No. 7, No. 39, No. 43 dataset. It is noteworthy that the accuracy of *MANB-BDeu* is the highest, although the value of "Missing variables" is somewhat large in the No. 42 dataset. The reason is that the Markov blanket includes important feature variables because the sample size for learning is sufficiently large. Generally speaking, the value of "Missing variables" tends to be small when the sample size is large such as those of the No. 12, No. 22, and No. 40 datasets. These results show that *MANB-BDeu* outperforms *ANB-BDeu* when the sample size is large.

6. Conclusions

First, this study compares the classification performances of the Bayesian networks exactly learned by BDeu and those approximately learned by CLL. Surprisingly, the results demonstrated that the

performance of Bayesian networks achieved by maximizing ML is not necessarily worse than that from maximizing CLL. However, the results also show that the classification accuracies of the Bayesian networks exactly learned by BDeu are much worse than those by the other methods when the class variable has numerous parents and a few children. To solve the problem, second, this study proposes an exact learning ANB by maximizing BDeu over the class variable's Markov blanket in the exact learned GBN. The experimentally obtained results show that the proposed method significantly outperforms approximately learned structure by maximizing CLL. However, the classification accuracy of the proposed method tends to lower when the Markov blanket feature selection discards numerous relevant variables to the class variable. Results show that the number of the discarded relevant variables tends to be small when the sample size is large. Therefore, the proposed method improves the classification accuracy when the sample size is large.

Our proposed method improves the performance of a Bayes classifier learned by BDeu when data are sparse. Recently, Scutari (2016, 2018) reported that BDeu should not be used for sparse data. They proposed a new Bayesian–Dirichlet sparse (BDs) score that provides better accuracy for sparse data. Therefore, if using the BDs score instead of the BDeu for exact learning GBN and ANB, then the classification accuracies might be improved.

References

W. Buntine. Theory Refinement on Bayesian Networks. In *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence*, pages 52–60, San Francisco, CA, USA, 1991. Morgan Kaufmann Publishers Inc.

A. M. Carvalho, P. Adão, and P. Mateus. Efficient Approximation of the Conditional Relative Entropy with Applications to Discriminative Learning of Bayesian Network Classifiers. *Entropy*, 15(7):2716–2735, 2013.

J. Cussens. Bayesian network learning with cutting planes. In *Proceedings of the 27th Conference on Uncertainty in Artificial Intelligence*, pages 153–160, 2012.

C. P. de Campos, M. Cuccu, G. Corani, and M. Zaffalon. *Extended Tree Augmented Naive Classifier*, pages 176–189. Springer International Publishing, Cham, 2014.

J. Demšar. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.*, 7: 1–30, 2006.

N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian Network Classifiers. *Machine Learning*, 29 (2):131–163, 1997.

R. Greiner and W. Zhou. Structural Extension to Logistic Regression: Discriminative Parameter Learning of Belief Net Classifiers. In *Eighteenth National Conference on Artificial Intelligence*, pages 167–173, 2002.

D. Grossman and P. Domingos. Learning Bayesian Network classifiers by maximizing conditional likelihood. In *Proceedings, Twenty-First International Conference on Machine Learning, ICML 2004*, pages 361–368, 2004.

D. Heckerman, D. Geiger, and D. M. Chickering. Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. *Machine Learning*, 20(3):197–243, 1995.

G. Hommel. A stagewise rejective multiple test procedure based on a modified bonferroni test. *Biometrika*, pages 383–386, 1988.

P. Langley and S. Sage. Induction of selective bayesian classifiers. In *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, pages 399–406, 1994.

C. X. Ling and H. Zhang. The representational power of discrete bayesian networks. *J. Mach. Learn. Res.*, 3:709–721, 2003.

P. J. F. Lucas. Restricted bayesian network structure learning. In *First European Workshop on Probabilistic Graphical Models, 6-8 November- 2002 - Cuenca (Spain), Electronic Proceedings*, 2002.

M. Minsky. Steps toward Artificial Intelligence. In *Proceedings of the IRE*, volume 49, pages 8–30, 1961.

M. Scutari. An empirical-bayes score for discrete bayesian networks. In *Probabilistic Graphical Models - Eightth Int. Conf. PGM 2016. Proceedings*, pages 438–448, 2016.

M. Scutari. Dirichlet bayesian network scores and the maximum relative entropy principle. *Behaviormetrika*, Apr 2018. ISSN 1349-6964. doi: 10.1007/s41237-018-0048-x.

T. Silander and P. Myllymäki. A Simple Approach for finding the Globally Optimal Bayesian Network Structure. In *Proceedings of Uncertainty in Artificial Intelligence*, pages 445–452, 2006.

T. Silander, P. Kontkanen, and P. Myllymäki. On sensitivity of the map bayesian network structure to the equivalent sample size parameter. In *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*, UAI'07, pages 360–367, 2007.

H. Steck. Learning the bayesian network structure: Dirichlet prior vs data. In *UAI 2008, Proceedings of the 24th Conference in Uncertainty in Artificial Intelligence, Helsinki, Finland, July 9-12, 2008*, pages 511–518, 2008.

J. Suzuki. A theoretical analysis of the BDeu scores in bayesian network structure learning. *Behaviormetrika*, 44(1):97–116, 2017.

M. Ueno. Learning likelihood-equivalence Bayesian networks using an empirical Bayesian approach. *Behaviormetrika*, 35(2):115–135, 2008.

M. Ueno. Learning Networks Determined by the Ratio of Prior and Data. In *Proceedings of Uncertainty in Artificial Intelligence*, pages 598–605, 2010.

M. Ueno. Robust learning Bayesian networks for prior belief. In *Proceedings of Uncertainty in Artificial Intelligence*, pages 689–707, 2011.

C. Yuan and B. Malone. Learning Optimal Bayesian Networks: A Shortest Path Perspective. *Journal of Artificial Intelligence Research*, 48(1):23–65, 2013.