


Attribute weighting for averaged one-dependence estimators

Zhong-Liang Xiang¹ · Dae-Ki Kang² 

Published online: 14 October 2016
© Springer Science+Business Media New York 2016

Abstract Averaged one-dependence estimators (AODE) is a type of supervised learning algorithm that relaxes the conditional independence assumption that governs standard naïve Bayes learning algorithms. AODE has demonstrated reasonable improvement in terms of classification performance when compared with a naïve Bayes learner. However, AODE does not consider the relationships between the super-parent attribute and other normal attributes. In this paper, we propose a novel method based on AODE that weighs the relationship between the attributes called weighted AODE (WAODE), which is an attribute weighting method that uses the conditional mutual information metric to rank the relations among the attributes. We have conducted experiments on University of California, Irvine (UCI) benchmark datasets and compared accuracies between AODE and our proposed learner. The experimental results in our paper show that WAODE exhibits higher accuracy performance than the original AODE.

Keywords Attribute weight · Structure extension · Conditional mutual information · Averaged one-dependence estimators · Bayesian model

✉ Dae-Ki Kang
dkkang@dongseo.ac.kr

Zhong-Liang Xiang
ugood@163.com

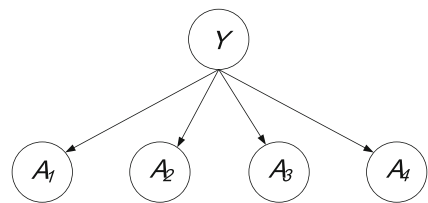
¹ Department of Computer Software Institute, Weifang University of Science and Technology, Shouguang 262-700, Shandong, China

² Department of Computer Engineering, Dongseo University, 47, Churye-Ro, Sasang-Gu, Busan, 47011, Republic of Korea

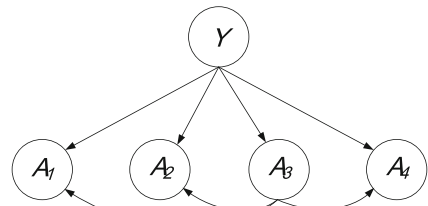
1 Introduction

A Bayesian network (BN) is also called a belief network and is a directed acyclic graphical model that belongs to the family of probabilistic graphical models. BN can present probabilistic relations among variables [8] and can present causal structures induced from datasets of regular attributes, in practice. However, learning the optimal BN structure from an arbitrary BN search space of discrete variables is considered non-deterministic polynomial time-hard (NP-hard) [2]. In [2], the author suggested that research should be directed away from the search for a general, exactly presented algorithm and toward the design of efficient special-case, average-case, and approximate-case algorithms. Many researchers have followed these suggestions and have proposed some relatively reasonable algorithms that restrict the complexity of the structures of BNs, such as naïve Bayes tree (NBTree) [11], tree-augmented naïve Bayes (TAN) [6], hidden naïve Bayes (HNB) [10], and averaged one-dependence estimators (AODE) [19]. Examples of the networks naïve Bayes, AODE, TAN, and HNB are shown in Fig. 1a, b, c, and d respectively. We will discuss the naïve Bayes learning algorithm at the beginning of this paper because it is efficient and has the simplest structure of a BN among the aforementioned learning algorithms based on BNs.

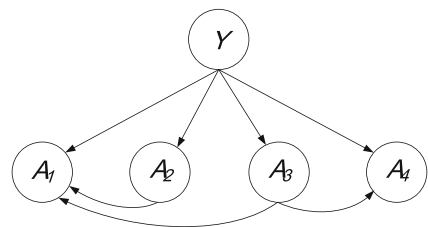
A naïve Bayes learner is a supervised learning method based on the Bayes rule. It runs on labeled training examples and is driven by the strong assumption that all attributes in the training examples are independent of each other given the class: the so-called naïve Bayes conditional independence assumption. Naïve Bayes has demonstrated high performance and rapid classification speed in huge



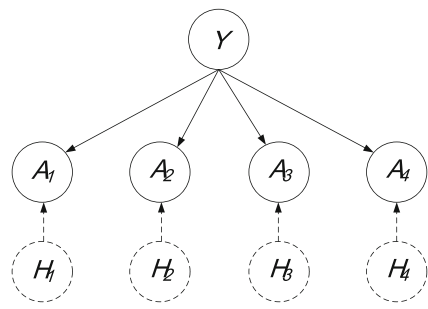
(a) Example of a naïve Bayes learner: Y is a class attribute and A_i is the i^{th} attribute in a dataset



(b) Example of AODE



(c) Example of TAN: Y is a class attribute and A_i is the i^{th} attribute in a dataset



(d) Example of HNB: Y is a class attribute, A_i is the i^{th} attribute in a dataset, and H_i (dashed line) represents a hidden parent node of A_i

Fig. 1 Bayesian net structures of naïve Bayes, AODE, TAN, and HNB

training instances with many attributes [14]. These benefits essentially originate from the naïve Bayes conditional independence assumption. In practice, classification accuracy could be affected by the conditional independence assumption, which is often violated in real-world applications. However, because of the attractive advantages of efficiency and simplicity, both originating from the class conditional independence assumption of attributes, many researchers have proposed effective methods to further improve the performance of naïve Bayes learners by weakening this strict attribute independence assumption without neglecting its

advantages. From the perspective of BN, the methods used to weaken the attribute independence assumption for the naïve Bayes learner would prefer to learn a BN's structure from a constrained BN search space rather than arbitrary BN search space.

Webb et al. [19] proposed a novel method called AODE. In AODE, the conditional probability of test instances given the class is tuned by a normal attribute (a super-parent of any other attribute). AODE has demonstrated significant improvement compared with naïve Bayes learners in two respects: The AODE classifier aggregates a set of weak classifiers and AODE relatively relaxes the independence assumption without learning the structure of a BN.

However, AODE does not consider the relationships between a super-parent node attribute, which is also called a *one-dependence* attribute in AODE, and other normal attributes. It is plausible that, for any pair of features A_i and A_j , the higher the dependence on each other, the more confidence there is in the probabilistic estimated values of $P(A_j|A_i)$ and $P(A_i|A_j)$. Hence, we propose weighted AODE (WAODE), which is an algorithm based on the AODE framework, and we expect it to improve the performance of AODE by considering the relation among features in real-world datasets. We expect that, based on our work, human experts can use their expertise to further tune the parameters that reflect the relationship among the pairs of attributes.

Zheng and Webb [21] proposed an improved AODE method called lazy elimination (LE) for AODE. LE is an interesting method that is simple, efficient, and can be applied to any Bayesian classifier. LE defines the *specialization-generalization* relationship between pairwise attributes and indicates that deleting all generalization attribute-values before training a classifier can relatively improve the performance of a Bayesian classifier. Zheng and Webb [21] proved that there is no information loss in the prediction after deleting the generalization attribute-values in the dataset. However, storing the generalization attribute-values (useless data) in the training datasets can cause prediction bias under the independence assumption because the independence assumption has the bias per se, which is violated in real-world problems.

Jiang and Zhang [9] proposed an algorithm called weightily averaged one-dependence estimators, which assigns weights to super-parent nodes according to the relationship between the super-parent node and class. The method proposed by [9] is different from our proposed method, WAODE, because the weight values in our paper indicate the relationships between the super-parent node and other normal attributes given the class.

Jiang et al. [10] proposed an algorithm called HNB, which is a structure extending method of naïve Bayes.

In HNB, each feature in a dataset has one hidden parent that is a mixture of the weighted influences from all other attributes. HNB and our WAODE apply the same attribute weighting method to measure the relationship between a pair of attributes using the conditional mutual information metric. Additionally, both HNB and WAODE avoid direct learning of the BN structure. However, WAODE is different from HNB in two respects: first, in HNB, the probability estimator of an attribute relies on the hidden parent of that attribute, but in WAODE, the probability estimator of an attribute depends on the training data sampling. In fact, this difference originates from the BN's structural difference between HNB and AODE. The second difference relates to the attribute weighing form: the weight is considered as a multiplier in the HNB learner, but as an exponent in WAODE.

Zaidi et al. [20] proposed a weighted naïve Bayes algorithm called Weighting attributes to Alleviate Naïve Bayes' Independence Assumption (WANBIA), that optimizes weights to minimize either the negative conditional log likelihood or the mean squared error objective functions. WAODE and its variants use simple functions of conditional mutual information as weight multipliers.

There are two contributions of this paper:

- We conduct a brief survey on improving AODE.
- We propose a novel attribute weighting for AODE called WAODE. WAODE not only retains the advantages of AODE, but also considers the dependence among the attributes using conditional mutual information measure methods.

Our experimental results show that the WAODE learning algorithm provides an effective improvement when compared with standard AODE.

The remainder of this paper is organized as follows: we briefly discuss AODE and attribute weighting forms in Section 2. In Section 3, we explain WAODE. In Section 4, we describe the experiments and results in detail. Finally, we draw the conclusions of our study and describe future work in Section 5.

2 AODE and attribute weighting forms

In this section, we provide a brief overview of AODE. AODE is a supervised learning algorithm based on a naïve Bayes learner, which extends the learner by a limited relaxation of the conditional independence assumption. Averaged one-attribute-dependence is the main feature of AODE. Because AODE is extended from a naïve Bayes learner, we briefly introduce the naïve Bayes learner first. At the final part of this section, we discuss the forms of attribute weighting.

2.1 Naïve Bayes learner

In supervised learning, consider a training dataset $\mathcal{D} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\}$ composed of n instances, where each instance $\mathbf{x} = \langle x_1, \dots, x_m \rangle \in \mathcal{D}$ (m -dimensional vector) is labeled with a class label $y \in Y$. For the posterior probability of y given by \mathbf{x} ,

$$p(y|\mathbf{x}) = \frac{p(\mathbf{x}|y) \cdot p(y)}{p(\mathbf{x})} \propto p(\mathbf{x}|y). \quad (1)$$

Note that it is usually difficult to estimate the likelihood $p(\mathbf{x}|y)$ directly from \mathcal{D} because there are insufficient data, in practice. Naïve Bayes uses the attribute independence assumption to alleviate this problem. From the assumption, $p(\mathbf{x}|y)$ can be estimated as follows:

$$p(\mathbf{x}|y) = p(x_1, \dots, x_m|y) \approx \prod_{i=1}^m p(x_i|y). \quad (2)$$

Then, the classifier of naïve Bayes is

$$\arg \max_{y \in Y} \hat{p}(y) \prod_{i=1}^m \hat{p}(x_i|y). \quad (3)$$

2.2 AODE

AODE structurally extends naïve Bayes by applying an averaged one-attribute-dependence method. The details of AODE are described as follows:

Given a test instance $\mathbf{x} = \langle x_1, \dots, x_i, \dots, x_m \rangle$, we have $p(y, \mathbf{x}) = p(y|\mathbf{x})p(\mathbf{x})$. Then

$$p(y|\mathbf{x}) = \frac{p(y, \mathbf{x})}{p(\mathbf{x})} = \frac{p(y)p(\mathbf{x}|y)}{p(\mathbf{x})}. \quad (4)$$

Assuming that there exists a training dataset with sufficient data, a test instance \mathbf{x} should be included in the training dataset. Because the attribute x_i is in \mathbf{x} ,

$$p(y, \mathbf{x}) = p(\mathbf{x}, x_i, y) = p(y, x_i)p(\mathbf{x}|y, x_i). \quad (5)$$

By combining (4) and (5),

$$p(y|\mathbf{x}) = \frac{p(y, x_i)p(\mathbf{x}|y, x_i)}{p(\mathbf{x})}. \quad (6)$$

Based on (6) and the attribute independence assumption, it follows that

$$p(\mathbf{x}|y, x_i) \approx \prod_{j=1}^m p(x_j|y, x_i), \quad (7)$$

where both x_j and x_i are the attribute-values in the test sample \mathbf{x} . Hence, the classifier of AODE can be described as follows:

$$\arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m \hat{p}(x_j|y, x_i) \right), \quad (8)$$

where $F(x_i)$ is the count of attribute-value x_i in the training set and g is the minimum sample size for meaningful statistical inference. Following [19], we set g to 30, which is one of the widely used values.

2.3 Attribute weighting forms

Because AODE is an extended learning algorithm based on naïve Bayes, we discuss our forms of attribute weighting based on naïve Bayes as preliminary background information.

Essentially, there are several formulations of naïve Bayes attribute weighting schemes. First, the weight for each attribute is defined as follows:

$$\hat{p}(c|\mathbf{x}) = \hat{p}(c) \prod_{i=1}^m \hat{p}(x_i|c)^{w_i}. \tag{9}$$

If the weight depends on an attribute and class, the corresponding formula is as follows:

$$\hat{p}(c|\mathbf{x}) = \hat{p}(c) \prod_{i=1}^m \hat{p}(x_i|c)^{w_{ci}}. \tag{10}$$

The following formula is used for the case in which the weight depends on an attribute value:

$$\hat{p}(c|\mathbf{x}) = \hat{p}(c) \prod_{i=1}^m \hat{p}(x_i|c)^{w_{i,x_i}}. \tag{11}$$

Referring back to (9), when $\forall w_i = w$, the formula is as follows:

$$\hat{p}(c|\mathbf{x}) = \hat{p}(c) \prod_{i=1}^m \hat{p}(x_i|c)^w. \tag{12}$$

Note that, in these weighting forms, if each weight is one, then each weighting form is simply naïve Bayes. For example, (12) is considered to be an extended version of the naïve Bayes classifier given that it will be the naïve Bayes classifier if each attribute A_i has the same weight $\forall w_i = w = 1$. Similarly, in (8), we can regard $p(x_j|y, x_i)$ of the AODE classifier to be analogous to $p(x_i|y)$ of naïve Bayes.

3 WAODE

In this section, we introduce our four forms of attribute weighting for AODE and then discuss the corresponding attribute-weight generating methods that can be used in AODE with our four proposed forms. Finally, we discuss our proposed WAODE algorithm.

3.1 Attribute weighting forms for AODE

In (8), we observe that the AODE learner does not consider the relation between attribute- j and attribute- i , which is also called the one-dependence attribute in AODE. Essentially, we propose using a set of parameters w_{ij} to indicate the relatively suitable relation values between the pairs of attributes i and j under a particular attribute weighting form in AODE.

Thus, we propose four forms of attribute weighting for AODE:

$$p(y, x_i) \prod_{j=1}^m (p(x_j|y, x_i) \cdot w_{ij}), \tag{13}$$

$$p(y, x_i) \prod_{j=1}^m p(x_j|y, x_i)^{w_{ij}}, \tag{14}$$

$$p(y, x_i) \prod_{j=1}^m p(x_j|y, x_i)^{\sqrt{w_{ij}}}, \tag{15}$$

and

$$p(y, x_i) \prod_{j=1}^m (p(x_j|y, x_i) \cdot \sqrt{w_{ij}}). \tag{16}$$

Note that each form of attribute weighting for AODE has a corresponding WAODE classifier shown as follows:

$$\arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m \hat{p}(x_j|y, x_i) \cdot w_{ij} \right), \tag{17}$$

$$\arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m \hat{p}(x_j|y, x_i)^{w_{ij}} \right), \tag{18}$$

$$\arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m \hat{p}(x_j|y, x_i)^{\sqrt{w_{ij}}} \right), \tag{19}$$

and

$$\arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m \hat{p}(x_j|y, x_i) \cdot \sqrt{w_{ij}} \right), \tag{20}$$

where w_{ij} is the weight value that describes the relation between attribute- i and attribute- j . The details of w_{ij} are

discussed in Section 3.2. The definitions of the remaining parameters in (13)–(20) were discussed in Section 2.

3.2 WAODE

Some methods to generate attribute weights have been used in attribute weighting for the naïve Bayes classifier. Lee et al. [13] generated feature weights using the Kullback–Leibler measurement. Orhan et al. [16] used the least squares approach to weight attributes in naïve Bayes. Zaidi et al. [20] proposed a weighted naïve Bayes algorithm called weighting attributes to alleviate the naïve independence assumption (WANBIA), which selects weights to maximize the conditional log-likelihood or minimize the mean squared error for the corresponding objective functions.

For AODE, we expect that the attribute weight in AODE should indicate the relation between a pair of attributes: w_{ij} should be assigned with a higher value when attribute- i , as a dependence attribute to attribute- j , provides more support to attribute- j and vice versa. Figure 2 shows the weighting form that our WAODE algorithm uses.

The conditional mutual information metric is used to generate this type of attribute weight. In our WAODE algorithm, weight w_{ij} , which is the weight value of the i^{th} attribute given the j^{th} attribute, is formulated as follows:

$$w_{ij} = \frac{I_P(A_i; A_j|Y)}{\sum_{i=1}^m \sum_{j=2, j>i}^m I_P(A_i; A_j|Y)} \cdot m, \tag{21}$$

where m is the number of attributes and $I_P(A_i; A_j|Y)$ is the conditional mutual information among attribute- i and attribute- j given class Y . $I_P(A_i; A_j|Y)$ is shown as follows:

$$I_P(A_i; A_j|Y) = \sum_{i,j,y} p(x_i, x_j, y) \log \frac{p(x_i, x_j|y)}{p(x_i|y)p(x_j|y)}, \tag{22}$$

where x_i and x_j are the attribute-values that correspond to the i^{th} attribute and j^{th} attribute, respectively. Note that $w_{ij} = w_{ji}$ because $I_P(A_i; A_j|Y) = I_P(A_j; A_i|Y)$.

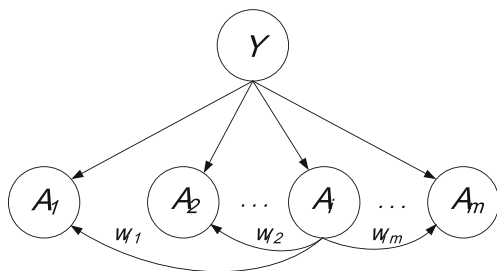


Fig. 2 The weighting model and structure of WAODE: Y is the class attribute and w_{ij} is a weight value of the pair of attributes A_i and A_j

Furthermore, the likelihood in AODE, $p(x_j|y, x_i)$, equals one when $i = j$.

$$\mathbf{W} = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix}. \tag{23}$$

Hence, we only need to calculate w_{ij} , such that $i = 1 \dots m$ and $j > i$. For example, consider a simple dataset with three attributes and (23) shows the matrix of attribute weights. In this case, we only need to calculate the upper (or lower) triangular part with the exception of the diagonal elements of the attribute matrix \mathbf{W} . We choose the red part of \mathbf{W} in this paper, which is shown in (23).

Now, we propose WAODE, whose attribute weights are generated by (21). The classifier of WAODE is shown as follows:

$$C_{WAODE}(\mathbf{t}) = \arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m (\hat{p}(x_j|y, x_i) \cdot w_{ij}) \right). \tag{24}$$

The learning algorithm of WAODE is shown in Fig. 3.

According to (18)–(20), we obtain the following classifiers, C_{WAODE1} , C_{WAODE2} , and C_{WAODE3} , respectively, which are shown as follows:

$$C_{WAODE1}(\mathbf{t}) = \arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m \hat{p}(x_j|y, x_i)^{w_{ij}} \right), \tag{25}$$

$$C_{WAODE2}(\mathbf{t}) = \arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m \hat{p}(x_j|y, x_i)^{\sqrt{w_{ij}}} \right), \tag{26}$$

and

$$C_{WAODE3}(\mathbf{t}) = \arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m \hat{p}(x_j|y, x_i) \cdot \sqrt{w_{ij}} \right). \tag{27}$$

During the training time, WAODE requires two cardinal time-consuming steps: first, to estimate $p(y)$, $p(x_i, y)$, and $p(x_i, x_j, y)$, it requires time complexity $O(tm^2)$. Additionally, compared with AODE, WAODE needs more time to

Fig. 3 Weighted averaged one-dependence estimators (WAODE) algorithm**WAODE:****Input:** training dataset $\mathcal{D} = \{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\}$ and a test instance $\mathbf{t} = \langle x_1, \dots, x_m \rangle$ **Output:** the class estimation of \mathbf{t} **Training phase:****begin**1. **Generating Weights:**(a) set $\forall w_{ij} = 1$, where $i = j$ (b) **for** each pair of attribute- i and j where $j > i$:

$$w_{ij} = \frac{I_P(A_i; A_j | Y)}{\sum_{i=1}^m \sum_{j=2, j>i}^m I_P(A_i; A_j | Y)} \cdot m$$

2. **Generating the tables of joint attribute-value:**(a) **for** each attribute, attribute-value and class-label y :calculate $p(y)$, $p(y, x_i)$ and $p(y, x_i, x_j)$ **end.****Classification phase:****begin**1. set $g = 30$ and filter x_i by $F(x_i) \geq g$.2. according to the test sample $\mathbf{t} = \langle x_1, \dots, x_m \rangle$, **for** each attribute- j and selected attribute- i :
calculate $p(x_j | y, x_i)$ 3. **Output** the class value:

$$C_{WAODE}(\mathbf{t}) = \arg \max_{y \in Y} \left(\sum_{i:1 \leq i \leq m \wedge F(x_i) \geq g} \hat{p}(y, x_i) \prod_{j=1}^m (\hat{p}(x_j | y, x_i) \cdot w_{ij}) \right)$$

end.

calculate weights for all pairs of attributes, which requires time complexity $O(kn^2v^2)$. The classification of a single example requires the calculation of (24) and is of time complexity $O(kn^2)$. Table 1 displays the time complexity of WAODE and the other algorithms discussed.

4 Evaluation

We conducted experiments on 25 datasets from the University of California, Irvine (UCI) Machine Learning Repository [1] to compare the performance of WAODE with that of naïve Bayes, AODE, NBTree, HNB, TAN, WANBIA^{CLL}, and the variations of WAODE: WAODE1, WAODE2, and WAODE3. The UCI benchmark datasets that we used are shown in Table 2. We discretized all numeric attributes to each dataset.

Table 1 Computational complexity, where k is the number of classes, n is the number of attributes, v is the average number of values for an attribute, and t is the number of training examples

Algorithm	Training time	Classification time
NB	$O(nt)$	$O(kn)$
AODE	$O(tn^2)$	$O(kn^2)$
TAN	$O(tn^2 + kn^2v^2 + n^2 \log n)$	$O(kn)$
HNB	$O(tn^2 + kn^2v^2)$	$O(kn^2)$
WAODE	$O(tn^2 + kn^2v^2)$	$O(kn^2)$

Table 2 Description of datasets used in the experiments

No.	Dataset	Instances	Attributes	Classes	Missing	Numeric
1	anneal	898	39	6	Y	Y
2	balance-scale	625	5	3	N	Y
3	breast-cancer	286	10	2	Y	N
4	breast-w	699	10	2	Y	N
5	credit-a	690	16	2	Y	Y
6	dermatology	366	35	6	Y	Y
7	diabetes	768	9	2	N	Y
8	glass	214	10	7	N	Y
9	heart-c	303	14	5	Y	Y
10	heart-h	294	14	5	Y	Y
11	heart-statlog	250	14	2	N	Y
12	hepatitis	155	20	2	Y	Y
13	hypothyroid	3772	30	4	Y	Y
14	ionosphere	351	35	3	N	Y
15	kr-vs-kp	3196	37	2	N	N
16	labor	57	12	2	Y	Y
17	letter	20000	17	26	N	Y
18	lymph	148	19	4	N	Y
19	primary-tumor	339	18	21	Y	N
20	sick	3772	30	2	Y	Y
21	sonar	208	61	2	N	Y
22	splice	3190	62	3	N	N
23	vehicle	846	19	4	N	Y
24	vote	435	17	2	Y	N
25	waveform-5000	5000	41	3	N	Y

Table 3 Experimental results in terms of classifier accuracies; accuracies are estimated using 10-fold cross-validation with a 95 % confidence interval

Data	Naïve Bayes	AODE	NBTree	HNB MI	TAN BIC	WANBIA CLL
(a) Results of algorithms except WAODE						
anneal	93.99±1.55	97.77±0.97	98.66±0.75	98.89±0.69	98.89±0.69	98.01±0.91
Balance-scale	91.36±2.20	89.60±2.39	91.36±2.20	90.08±2.34	86.56±2.67	74.06±3.44
Breast-cancer	71.68±5.22	71.68±5.22	70.98±5.26	N/A	69.58±5.33	N/A
Breast-w	97.28±1.21	97.00±1.27	96.57±1.35	N/A	95.14±1.59	96.57±1.35
Credit-a	85.94±2.59	87.54±2.46	86.38±2.56	N/A	87.25±2.49	86.20±2.57
Dermatology	97.81±1.50	98.09±1.40	97.81±1.50	97.27±1.67	96.72±1.82	98.10±1.40
Diabetes	77.47±2.95	76.04±3.02	77.47±2.95	75.65± 3.04	76.43±3.00	78.49±2.91
Glass	77.10±5.63	78.97±5.46	71.96±6.02	74.30±5.85	78.50±5.50	82.43±5.10
Heart-c	85.81±3.93	84.82±4.04	84.82±4.04	N/A	83.17±4.21	N/A
Heart-h	86.39±3.92	86.05±3.96	85.17±4.00	N/A	86.39±3.92	82.99±4.29
Heart-statlog	83.70±4.41	84.07±4.36	82.59±4.52	84.44±4.32	82.22±4.56	N/A
Hepatitis	89.03±4.92	89.68±4.79	85.81±5.49	N/A	87.74±5.16	85.48±5.55
Hypothyroid	95.68±0.65	95.86±0.64	95.84±0.64	N/A	95.84±0.64	99.17±0.29
Ionosphere	92.02±2.83	93.45±2.59	92.31±2.79	93.45±2.59	94.02±2.48	92.21±2.80
Kr-vs-kp	87.89±1.13	91.24±0.98	97.09±0.58	92.46±0.92	92.05±0.94	93.04±0.88
Labor	91.23±7.34	82.46±9.87	87.72±8.52	N/A	84.21±9.47	94.56±5.89
Letter	72.84±0.62	85.20±0.49	83.09±0.52	86.46±0.47	84.20±0.51	75.16±0.60
Lymph	85.81±5.62	85.81±5.62	84.46±5.84	83.78±5.94	82.43±6.13	81.99±6.19
Primary-tumor	50.15±5.32	49.56±5.32	46.02±5.31	N/A	45.13±5.30	N/A
Sick	97.48±0.50	97.51±0.50	97.56±0.49	N/A	97.64±0.48	97.37±0.51
Sonar	99.04±1.33	99.04±1.33	99.04±1.33	100.00±0.00	96.15±2.61	83.12±5.09
Splice	95.36±0.73	96.21±0.66	95.36±0.73	96.18±0.67	95.49±0.72	95.89±0.69
Vehicle	66.67±3.18	73.52±2.97	72.22±3.02	76.95±2.84	76.83±2.84	67.11±3.17
Vote	90.11±2.80	94.25±2.19	95.63±1.92	N/A	94.94±2.06	95.48±1.95
Waveform-5000	64.04±1.33	64.02±1.33	64.04±1.33	63.78±1.33	63.34±1.34	N/A
Mean	85.04±2.94	85.98±2.95	85.60±2.95	86.69±2.33	85.23±3.06	87.87±2.78
(b) Results of the WAODE algorithms						
Data	WAODE1	WAODE2	WAODE3	WAODE(MI)		
Anneal	76.17±2.79	84.74±2.35	97.44±1.03	97.44±1.03		
Balance-scale	91.04±2.24	90.08±2.34	89.60±2.39	89.60±2.39		
Breast-cancer	72.38±5.18	74.13±5.08	73.08±5.14	72.03±5.20		
Breast-w	95.99±1.45	97.14±1.24	97.00±1.27	97.00±1.27		
Credit-a	84.49±2.70	86.67±2.54	87.39±2.48	87.10±2.50		
Dermatology	93.17±2.58	97.81±1.50	97.81±1.50	97.81±1.50		
Diabetes	69.92±3.24	75.39±3.05	76.04±3.02	76.04±3.02		
Glass	71.96±6.02	74.30±5.85	78.97±5.46	78.97±5.46		
Heart-c	84.49±4.08	84.16±4.11	84.82±4.04	85.48±3.97		
Heart-h	83.33±4.26	83.67±4.22	85.37±4.04	85.37±4.04		
Heart-statlog	81.85±4.60	83.70±4.41	83.33±4.45	83.70±4.41		
Hepatitis	79.35±6.37	89.03±4.92	89.03±4.92	89.68±4.79		
Hypothyroid	92.29±0.85	92.29±0.85	95.89±0.63	95.89±0.63		
Ionosphere	90.31±3.09	92.88±2.69	92.02±2.83	93.45±2.59		
Kr-vs-kp	79.72±1.39	87.67±1.14	91.33±0.98	91.27±0.98		
Labor	91.23±7.34	89.47±7.97	85.96±9.02	85.96±9.02		
Letter	77.71±0.58	80.89±0.54	85.20±0.49	85.20±0.49		
Lymph	81.08±6.31	85.14±5.73	87.16±5.39	86.49±5.51		
Primary-tumor	29.50±4.85	47.20±5.31	49.85±5.32	50.15±5.32		
Sick	93.88±0.76	93.88±0.77	97.38±0.51	97.38±0.51		
Sonar	88.94±4.26	99.04±1.33	99.04±1.33	99.04±1.33		
Splice	52.04±1.73	91.76±0.95	95.71±0.70	95.67±0.71		
Vehicle	69.62±3.10	71.39±3.05	73.52±2.97	73.52±2.97		
Vote	90.57±2.75	92.87±2.42	94.25±2.19	94.25±2.19		
Waveform-5000	63.92±1.33	64.06±1.33	64.02±1.33	64.02±1.33		
Mean	79.40±3.36	84.37±3.03	86.05±2.94	86.10±2.93		

The number in boldface indicates the algorithm with the highest performance

In the implementation of our algorithm, all the estimated probabilities, including $\hat{p}(y)$, $\hat{p}(x_i, y)$, and $\hat{p}(x_i, x_j, y)$, were filtered using Laplacian smoothing estimations because of the zero count of the probabilistic calculation. The details are as follows:

$$\hat{p}(y) = \frac{F(y) + 1}{n + c(Y)}, \tag{28}$$

$$\hat{p}(x_i, y) = \frac{F(x_i, y) + 1}{n_i + c(attri_i) \times c(Y)}, \tag{29}$$

$$\hat{p}(x_i, x_j, y) = \frac{F(x_i, x_j, y) + 1}{n_{ij} + c(attri_i) \times c(attri_j) \times c(Y)}, \tag{30}$$

where n is the number of training examples for which the class value is known; n_i is the number of training examples for which both attribute i and the class are known; n_{ij} is the number of training examples for which all the classes, and attributes i and j are known; $c(\bullet)$ is the cardinality value of \bullet ; and $F(\bullet)$ is the frequency of \bullet .

We conducted experiments on the proposed algorithms, standard naïve Bayes, AODE [19], NBTree [11], HNB [10], and TAN [6]. To compare the performance of the algorithms, we used a t -test with 10-fold cross-validation. We applied all algorithms to the same training datasets and test datasets. We directly copied the experimental results of WANBIA^{CLL} from [20].

Note that there are missing results for some datasets for HNB because the datasets have missing values. Additionally, there are missing results for WANBIA^{CLL} that are not in [20].

Table 3 shows the accuracy results of WAODE, variations of WAODE, and previous developed algorithms on 25 UCI benchmark datasets. The average accuracies and 95 % confidence interval on all datasets are summarized at the bottom of the table. Conditional mutual information was used for WAODE, WAODE1, WAODE2, WAODE3, and HNB. Bayesian information criteria (BIC)

were used for TAN. Conditional log-likelihood was used for WANBIA^{CLL}. Some results are missing for the HNB algorithm because HNB cannot effectively manage datasets with missing values. Instead of using traditional methods to fill in the missing values or considering each missing value as a new attribute value, we simply omitted HNB from the experiments of datasets with missing values for a fair comparison. The results of dataset “breast-cancer,” “heart-c,” “heart-statlog,” “primary-tumor,” and “waveform-5000” are missing for WANBIA^{CLL} because we did not locate the results in the paper. For a fair comparison, we omitted the results of the five datasets mentioned above for WANBIA^{CLL}.

In Table 3, we observe that the classification performance of WAODE is better than all the other algorithms regarding the mean of the accuracy. Both WAODE1 and WAODE2, in which the weights are the exponential in the classifier, demonstrate worse results when compared with the other algorithms. WAODE3, which is same with WAODE, the weight as a multiplier in its classifier, demonstrates relatively better performance than WAODE1 and WAODE2.

Win(w)/tie(t)/lose(l) are summarized in Table 4. We observe that WAODE provides comparable and sometimes better performance than naïve Bayes with a win/tie/lose record of 13/4/8. WAODE only demonstrates slightly better performance, with a win/tie/lose record of 8/10/7 when compared with AODE, and 11/0/9, when compared with WANBIA^{CLL}.

Despite the missing results for HNB and WANBIA^{CLL}, WAODE did not outperform them for several datasets. We checked the results and found that the training accuracy of WAODE was usually higher than that of HNB. This implies that over-fitting degraded the performance of WAODE. However, WAODE can be used more broadly than HNB because WAODE can manage missing values more properly than HNB. Additionally, because we directly copied the experimental results of WANBIA^{CLL} from [20], our comparison in this paper may not be fair and accurate.

Table 4 Win/tie/lose results of standard naïve Bayes, AODE, NBTree, HNB, TAN, and WANBIA^{CLL}

win/tie/lose	Naïve Bayes	AODE	WAODE1	WAODE2	WAODE3	NBTree	HNB	TAN	WANBIA ^{CLL}
AODE	14/3/8	—	—	—	—	—	—	—	—
WAODE1	4/1/20	3/0/22	—	—	—	—	—	—	—
WAODE2	7/4/14	5/0/20	20/2/3	—	—	—	—	—	—
WAODE3	11/4/10	6/10/9	23/0/2	15/3/7	—	—	—	—	—
NBTree	8/7/9	8/2/14	22/0/2	12/2/10	10/3/11	—	—	—	—
HNB	9/0/5	8/1/5	12/0/2	10/1/3	10/0/4	7/0/6	—	—	—
TAN	11/1/13	9/0/16	19/0/6	13/0/12	8/0/17	9/1/14	2/1/11	—	—
WANBIA ^{CLL}	13/0/7	8/0/12	15/0/5	10/0/10	10/0/10	6/0/12	4/0/8	9/0/11	—
WAODE(MI)	13/4/8	8/10/7	22/0/3	17/3/5	6/14/5	13/2/9	4/1/9	16/0/9	11/0/9

Table 5 Experimental results in terms of classifiers' bias²

Data	Naïve Bayes	AODE	NBTree	HNB MI	TAN BIC
(a) Bias results of algorithms except WAODE					
Anneal	0.1529	0.1520	0.0912	0.0698	0.0717
Balance-scale	0.1345	0.1653	0.1473	0.1858	0.1875
Breast-cancer	0.2345	0.2381	0.2204	N/A	0.2144
Breast-w	0.0259	0.0296	0.0259	N/A	0.0416
Credit-a	0.1447	0.1476	0.1133	N/A	0.1320
Dermatology	0.0575	0.0534	0.0569	0.0476	0.0605
Diabetes	0.1899	0.1975	0.1920	0.2183	0.1906
Glass	0.2136	0.2159	0.1773	0.1655	0.1845
Heart-c	0.1285	0.1272	0.1135	N/A	0.1205
Heart-h	0.1488	0.1465	0.1443	N/A	0.1425
Heart-statlog	0.1414	0.1554	0.1405	0.1494	0.1433
Hypothyroid	0.0791	0.0777	0.0662	N/A	0.0772
Ionosphere	0.0835	0.0820	0.0722	0.0686	0.0732
Kr-vs-kp	0.1991	0.1803	0.1024	0.1503	0.1272
Letter	0.4882	0.4807	0.4455	0.4550	0.4307
Primary-tumor	0.3908	0.3926	0.3827	N/A	0.4003
Sick	0.0606	0.0604	0.0469	N/A	0.0385
Sonar	0.0429	0.0526	0.0307	0.0130	0.0463
Splice	0.1240	0.1334	0.1338	0.1250	0.2253
Vehicle	0.2739	0.2521	0.2230	0.2166	0.2424
Vote	0.0838	0.0479	0.0483	N/A	0.0439
Waveform-5000	0.3905	0.3841	0.2659	0.2708	0.2632
Average	0.1722	0.1715	0.1473	0.1643	0.1572
(b) Bias results of WAODE algorithms					
Data	WAODE1	WAODE2	WAODE3	WAODE(MI)	
Anneal	0.2381	0.2245	0.1534	0.1534	
Balance-scale	0.1508	0.1563	0.1653	0.1653	
Breast-cancer	0.2868	0.2621	0.2315	0.2288	
Breast-w	0.0555	0.0340	0.0294	0.0290	
Credit-a	0.1734	0.1570	0.1447	0.1450	
Dermatology	0.1994	0.0776	0.0646	0.0712	
Diabetes	0.2613	0.2148	0.1975	0.1975	
Glass	0.3003	0.2469	0.2159	0.2159	
Heart-c	0.1447	0.1321	0.1281	0.1276	
Heart-h	0.1912	0.1603	0.1462	0.1478	
Heart-statlog	0.2008	0.1633	0.1531	0.1509	
Hypothyroid	0.0779	0.0779	0.0777	0.0778	
Ionosphere	0.1050	0.0935	0.0834	0.0835	
Kr-vs-kp	0.1989	0.1866	0.1817	0.1818	
Letter	0.5009	0.4872	0.4807	0.4807	
Primary-tumor	0.5016	0.4048	0.3919	0.3913	
Sick	0.0610	0.0610	0.0604	0.0605	
Sonar	0.1445	0.0746	0.0526	0.0526	
Splice	0.3638	0.1325	0.1472	0.1525	
Vehicle	0.2493	0.2523	0.2521	0.2521	
Vote	0.0704	0.0600	0.0479	0.0480	
Waveform-5000	0.4158	0.4030	0.3841	0.3841	
Average	0.2223	0.1847	0.1722	0.1726	

The number in boldface indicates the algorithm with the highest performance

Table 6 Experimental results in terms of classifiers' variance

Data	Naïve Bayes	AODE	NBTree	HNB MI	TAN BIC
(a) Variance results of algorithms except WAODE					
Anneal	0.0286	0.0278	0.0533	0.0365	0.0280
Balance-scale	0.1165	0.1283	0.1473	0.1572	0.1796
Breast-cancer	0.0514	0.0510	0.0722	N/A	0.0933
Breast-w	0.0058	0.0151	0.0058	N/A	0.0524
Credit-a	0.0305	0.0288	0.0473	N/A	0.0626
Dermatology	0.0760	0.0751	0.1093	0.0685	0.1036
Diabetes	0.1065	0.1122	0.1322	0.1339	0.1253
Glass	0.0876	0.0867	0.1302	0.1007	0.1179
Heart-c	0.0407	0.0441	0.0755	N/A	0.0851
Heart-h	0.0245	0.0318	0.0537	N/A	0.0554
Heart-statlog	0.0400	0.0443	0.0710	0.0715	0.0897
Hypothyroid	0.0018	0.0001	0.0109	N/A	0.0009
Ionosphere	0.0099	0.0122	0.0183	0.0176	0.0262
Kr-vs-kp	0.0829	0.0870	0.1312	0.0870	0.1114
Letter	0.3259	0.3126	0.3666	0.3038	0.3675
Primary-tumor	0.2190	0.2170	0.2708	N/A	0.2826
Sick	0.0017	0.0006	0.0110	N/A	0.0126
Sonar	0.0707	0.0747	0.0828	0.0275	0.0715
Splice	0.1220	0.1446	0.2171	0.1259	0.2644
Vehicle	0.1395	0.1468	0.1797	0.1514	0.1715
Vote	0.0075	0.0108	0.0251	N/A	0.0309
Waveform-5000	0.2090	0.2131	0.2141	0.1984	0.2279
Average	0.0817	0.0848	0.1102	0.1138	0.1164
Data	WAODE1	WAODE2	WAODE3	WAODE(MI)	
(b) Variance results of WAODE algorithms					
Anneal	0.000	0.0062	0.0279	0.0279	
Balance-scale	0.1083	0.1153	0.1283	0.1283	
Breast-cancer	0.0160	0.0332	0.0500	0.0530	
Breast-w	0.0151	0.0119	0.0149	0.0144	
Credit-a	0.0478	0.0282	0.0285	0.0286	
Dermatology	0.0989	0.0893	0.0747	0.0731	
Diabetes	0.0636	0.0980	0.1122	0.1122	
Glass	0.0940	0.0917	0.0867	0.0867	
Heart-c	0.0464	0.0435	0.0436	0.0443	
Heart-h	0.0515	0.0371	0.0303	0.0290	
Heart-statlog	0.0468	0.0460	0.0428	0.0454	
Hypothyroid	0.0000	0.0000	0.0001	0.0001	
Ionosphere	0.0071	0.0072	0.0104	0.0099	
Kr-vs-kp	0.1714	0.1037	0.0868	0.0868	
Letter	0.3321	0.3183	0.3126	0.3126	
Primary-tumor	0.1640	0.1995	0.2177	0.2175	
Sick	0.0000	0.0000	0.0006	0.0006	
Sonar	0.1426	0.0951	0.0747	0.0747	
Splice	0.0606	0.1287	0.1583	0.1630	
Vehicle	0.1872	0.1563	0.1468	0.1468	
Vote	0.0135	0.0130	0.0108	0.0109	
Waveform-5000	0.2082	0.2076	0.2131	0.2131	
Average	0.0852	0.0832	0.0851	0.0854	

The number in boldface indicates the algorithm with the highest performance

Table 7 Experimental results in terms of the weighted average of AUC: average AUCs are estimated using 10-fold cross-validation with a 95 % confidence interval

Data	Naïve Bayes	AODE	NBTree	HNB MI	TAN BIC
(a) Results of the algorithms except WAODE					
Anneal	0.994±0.01	0.997±0.00	0.991±0.01	1.000±0.00	0.999±0.00
Balance-scale	0.962±0.01	0.949±0.02	0.962±0.01	0.966±0.01	0.941±0.02
Breast-cancer	0.701±0.05	0.718±0.05	0.691±0.05	N/A	0.665±0.05
Breast-w	0.993±0.01	0.993±0.01	0.991±0.01	N/A	0.986±0.01
Credit-a	0.930±0.02	0.939±0.02	0.924±0.02	N/A	0.942±0.02
Dermatology	0.999±0.00	0.999±0.00	0.996±0.01	0.999±0.00	0.999±0.00
Diabetes	0.844±0.03	0.836±0.03	0.844±0.03	0.826±0.03	0.830±0.03
Glass	0.932±0.03	0.942±0.03	0.926±0.04	0.949±0.03	0.940±0.03
Heart-c	0.931±0.03	0.931±0.03	0.931±0.03	N/A	0.924±0.03
Heart-h	0.927±0.03	0.926±0.03	0.921±0.03	N/A	0.919±0.03
Heart-statlog	0.922±0.03	0.921±0.03	0.906±0.04	0.924±0.03	0.907±0.04
Hepatitis	0.951±0.03	0.954±0.03	0.912±0.04	N/A	0.941±0.04
Hypothyroid	0.899±0.01	0.906±0.01	0.895±0.01	N/A	0.904±0.01
Ionosphere	0.978±0.02	0.988±0.01	0.972±0.02	0.993±0.01	0.987±0.01
Kr-vs-kp	0.952±0.01	0.974±0.01	0.993±0.00	0.982±0.00	0.981±0.00
Labor	0.965±0.05	0.936±0.06	0.930±0.07	N/A	0.900±0.08
Letter	0.976±0.00	0.995±0.00	0.984±0.00	0.996±0.00	0.993±0.00
Lymph	0.919±0.04	0.931±0.04	0.925±0.04	0.920±0.04	0.920±0.04
Primary-tumor	0.835±0.04	0.837±0.04	0.817±0.04	N/A	0.819±0.04
Sick	0.962±0.01	0.963±0.01	0.956±0.01	N/A	0.960±0.01
Sonar	1.000±0.00	1.000±0.00	1.000±0.00	1.000±0.00	0.997±0.01
Splice	0.993±0.00	0.995±0.00	0.993±0.00	0.995±0.00	0.992±0.00
Vehicle	0.874±0.02	0.931±0.02	0.892±0.02	0.939±0.02	0.933±0.02
Vote	0.973±0.02	0.990±0.01	0.985±0.01	N/A	0.986±0.01
Waveform-5000	0.813±0.01	0.813±0.01	0.813±0.01	0.816±0.01	0.814±0.01
Mean	0.929±0.02	0.935±0.02	0.926±0.022	0.950±0.01	0.927±0.02
Data	WAODE1	WAODE2	WAODE3	WAODE(MI)	
(b) Results of the WAODE algorithms					
Anneal	0.985±0.01	0.995±0.00	0.996±0.00	0.996±0.00	
Balance-scale	0.949±0.02	0.949±0.02	0.949±0.02	0.949±0.02	
Breast-cancer	0.714±0.05	0.713±0.05	0.714±0.05	0.717±0.05	
Breast-w	0.993±0.01	0.993±0.01	0.992±0.01	0.992±0.01	
Credit-a	0.928±0.02	0.934±0.02	0.942±0.02	0.942±0.02	
Dermatology	0.998±0.00	0.999±0.00	0.999±0.00	0.999±0.00	
Diabetes	0.837±0.03	0.838±0.03	0.836±0.03	0.836±0.03	
Glass	0.932±0.03	0.937±0.03	0.942±0.03	0.942±0.03	
Heart-c	0.925±0.03	0.926±0.03	0.931±0.03	0.931±0.03	
Heart-h	0.924±0.03	0.926±0.03	0.924±0.03	0.924±0.03	
Heart-statlog	0.914±0.03	0.917±0.03	0.918±0.03	0.920±0.03	
Hepatitis	0.923±0.04	0.937±0.04	0.955±0.03	0.956±0.03	
Hypothyroid	0.901±0.01	0.901±0.01	0.904±0.01	0.904±0.01	
Ionosphere	0.976±0.02	0.982±0.01	0.978±0.02	0.981±0.01	
Kr-vs-kp	0.961±0.01	0.962±0.01	0.975±0.01	0.975±0.01	
Labor	0.961±0.05	0.959±0.05	0.942±0.06	0.942±0.06	
Letter	0.976±0.00	0.989±0.00	0.995±0.00	0.995±0.00	
Lymph	0.925±0.04	0.927±0.04	0.928±0.04	0.928±0.04	
Primary-tumor	0.823±0.04	0.833±0.04	0.836±0.04	0.836±0.04	
Sick	0.959±0.01	0.966±0.01	0.962±0.01	0.962±0.01	
Sonar	0.999±0.00	1.000±0.00	1.000±0.00	1.000±0.00	
Splice	0.990±0.00	0.994±0.00	0.995±0.00	0.995±0.00	
Vehicle	0.888±0.02	0.914±0.02	0.931±0.02	0.931±0.02	
Vote	0.978±0.01	0.984±0.01	0.990±0.01	0.991±0.01	
Waveform-5000	0.795±0.01	0.810±0.01	0.813±0.01	0.813±0.01	
Mean	0.926±0.02	0.931±0.02	0.934±0.02	0.934±0.02	

The number in boldface indicates the algorithm with the highest performance

Compared with NBTree and TAN, WAODE exhibits comparable and sometimes better performance, with win/tie/lose records of 13/2/9 and 16/0/9, respectively.

For the detailed analysis, we have performed bias-variance decomposition [12, 18] results to compare the performance of the classifiers with varying amount of data. Following [12], we calculated bias² and variance using 2-fold cross-validation and listed the results in Tables 5 and 6 respectively.

Bias² shows the quadratic loss which is calculated by squared difference between the average of actual output and the average of predicted output. Variance is a non-negative value that reflects the variability to measure how much the learner is sensitive to changes in the training set. Variance is zero if the learner makes consistent prediction regardless of the training set. The results of datasets “hepatitis”, “labor” and “lymph” are not included because they have too few instances (less than 200 instances) for statistically significant bias-variance decomposition.

It can be seen that NBTree shows the least bias and naïve Bayes shows the smallest variance among the learning algorithms. This indicates that NBTree might exhibit relatively low error for large data and naïve Bayes might produce relatively low error for small data. Note that we do not include the bias and variance results of WANBIA^{CLL} from [20] because their paper has different experimental settings and datasets. WAODE(MI) algorithm and its variants do not show remarkable superiority or inferiority compared with other algorithms in terms of bias and variance. In Table 8, we will discuss these results in more detail with appropriate statistical tests.

The receiver operating characteristic (ROC) curve [5] is an effective tool to illustrate the behavior of binary classifiers.

The ROC curve is widely used in binary classification because it does not degrade itself on imbalanced datasets. The area under the ROC curve (AUC) reflects the probability that the classifier ranks a random positive instance higher than a random negative instance. AUC can be formulated as a Gini index minus one divided by two [7]. AUC is usually an effective measure but is not always a more valued measure than accuracy because of several drawbacks [7, 15]. Despite those drawbacks, the ROC curve and its AUC are still widely used for measuring classifiers’ performance.

One main drawback of the ROC curve for measuring classifiers’ performance is that it is usually effective for binary classification. Because most benchmark datasets we have tested are multiclass datasets, it is difficult to measure performance with AUC for multiclass classifiers. In Table 7, we list the weighted averages of the AUCs of WAODE and the remaining algorithms. Because the ROC curve and AUC can be calculated for binary classification, we followed a one-vs-rest approach for the AUC calculations.

From the results of Table 7, it can be seen that the overall results are somewhat mixed. WAODE(MI) shows much slightly lower or similar performance than AODE within the confidence interval. Again HNB shows slightly higher performance than AODE and WAODE(MI). As aforementioned, it is because over-fitting of WAODE and its variants reflected from its high training accuracy. Overall, we can see that AUC is generally too high for most classification results to effectively discriminate the classifiers’ performance, which complies the arguments of [7, 15].

For the statistical comparison of the classifiers, we employ binomial sign test and compare WAODE(MI) with other algorithms in Table 8. Following [19], we have calculated one-tailed p value for Naïve Bayes, and for other

Table 8 Win/Tie/Loss records of WAODE(MI) and other algorithms with binomial sign test

	Win/Tie/Loss	p-value	Win/Tie/Loss	p-value	Win/Tie/Loss	p-value
	Naïve Bayes		AODE		NBTree	
Accuracy	13/4/8	0.1917	7/11/7	1.2095	15/2/8	0.2100
Bias	10/1/11	0.5000	5/7/10	0.3018	1/0/21	< 0.0001
Variance	8/1/13	0.1917	6/9/7	1.0000	21/0/1	< 0.0001
AUC	14/5/6	0.0577	5/10/10	0.3018	19/3/3	0.0009
	HNB		TAN		WANBIA ^{CLL}	
Accuracy	5/1/8	0.2668	16/0/9	0.2295	11/0/9	0.8238
Bias	2/0/11	0.0225	4/0/18	0.0043	—	—
Variance	8/0/5	0.5811	21/0/1	< 0.0001	—	—
AUC	2/3/9	0.0654	17/3/5	0.0169	—	—

algorithms, we have reported two-tailed p values. Overall, WAODE(MI) shows higher performance than naïve Bayes. Compared with AODE, WAODE(MI) shows comparable performance with statistically approvable intervals. WAODE(MI) outperforms NBTree and TAN in accuracy, variance and AUC, but WAODE(MI) exhibits that it has high bias compared with them. As we noted before, overall, HNB shows higher performance than WAODE(MI). However, HNB does not always outperform WAODE(MI) and, considering the results with HNB in Table 8, reducing the bias error and avoiding over-fitting using proper regularization might lead WAODE(MI) to a promising direction.

5 Conclusion

In this paper, we proposed a novel Bayesian model called WAODE. Using our method, we considered the relationships between the super-parent attribute and other attributes. We conducted a systematic experiment to measure the classification accuracies of our WAODE algorithm, naïve Bayes, AODE, NBTree, HNB, TAN, WANBIA^{CLL}, and three variations of WAODE (WAODE1, WAODE2, and WAODE3) on 25 UCI benchmark datasets. The experimental results showed that WAODE exhibited higher performance when compared with naïve Bayes, AODE, and the other algorithms. We observed that, in WAODE, the design of the weight was the core issue. The quality of the weight dominated the classification performance of WAODE. We used the conditional mutual information metric to generate weights in WAODE.

In our future work, we will explore more effective methods to estimate weights from data to improve our WAODE performance. Furthermore, note that we currently consider the relationship between the parent node and child node only to generate weights. It will be interesting if we may add other factors in generating weights. For example, a hybrid weighting scheme may possibly enhance the relationship between the child nodes. The experimental results indicate that WAODE has high bias value compared with NBTree and TAN. Finding an optimal trade-off point between bias and variance will be an essential problem of WAODE. Additionally, we will further compare the performance of our algorithms with ensemble learning methods [3, 4, 17].

Acknowledgments This work was supported by the Korea Creative Content Agency (KOCCA) grant funded by the Korea government (no. R2015090021).

References

1. Bache K, Lichman M (2013) UCI machine learning repository
2. Cooper GF (1990) The computational complexity of probabilistic inference using Bayesian belief networks. *Artif Intell* 42(2):393–405
3. Dai Q (2013) A competitive ensemble pruning approach based on cross-validation technique. *Knowl-Based Syst* 37:394–414
4. Dai Q, Zhang T, Liu N (2015) A new reverse reduce-error ensemble pruning algorithm. *Appl Soft Comput* 28:237–249
5. Fawcett T (2006) An introduction to ROC analysis. *Pattern Recogn Lett* 27(8):861–874
6. Friedman N, Geiger D, Goldszmidt M (1997) Bayesian network classifiers. *Mach Learn* 29(2–3):131–163
7. Hand DJ (2009) Measuring classifier performance: a coherent alternative to the area under the ROC curve. *Mach Learn* 77(1):103–123
8. Heckerman D (2008) A tutorial on learning with Bayesian networks. In: *Innovations in Bayesian networks*. Springer, pp 33–82
9. Jiang L, Zhang H (2006) Weightily averaged one-dependence estimators. In: *The 9th pacific rim international conference on artificial intelligence (PRICAI 2006: trends in artificial intelligence)*. Springer, pp 970–974
10. Jiang L, Zhang H, Cai Z (2009) A novel Bayes model: hidden naïve bayes. *IEEE Trans Knowl Data Eng* 21(10):1361–1371
11. Kohavi R (1996) Scaling up the accuracy of naïve-Bayes classifiers: a decision-tree hybrid. In: *Proceedings of the 2nd international conference on knowledge discovery and data mining (KDD-96)*, pp 202–207
12. Kohavi Ron, Wolpert DavidH. (1996) Bias plus variance decomposition for zero-one loss functions. In: Saitta L (ed) *Proceedings of the 13th international conference machine learning*. Morgan Kaufmann, pp 275–283
13. Lee C-H, Gutierrez F, Dou D (2011) Calculating feature weights in naïve Bayes with kullback-leibler measure. In: *Proceedings of the 11th IEEE international conference on data mining (ICDM 2011)*. IEEE, pp 1146–1151
14. Lewis DD (1998) Naive (bayes) at forty: The independence assumption in information retrieval. In: *Proceedings of the 10th European conference on machine learning. ECML '98*. Springer, London, pp 4–15. ISBN 3-540-64417-2
15. Lobo JM, Jiménez-Valverde A, Real R (2007) AUC: a misleading measure of the performance of predictive distribution models. *Glob Ecol Biogeogr* 17(2)
16. Orhan U, Adem K, Comert O (2012) Least squares approach to locally weighted naïve Bayes method. *J New Results Sci* 1(1)
17. Partalas I, Tsoumakas G, Vlahavas I (2010) An ensemble uncertainty aware measure for directed hill climbing ensemble pruning. *Mach Learn* 81(3):257–282
18. Webb GI, Conilione P (2002) *Estimating bias and variance from data*. School of Computer Science and Software Engineering, Victoria
19. Webb GI, Boughton JR, Wang Z (2005) Not so naïve Bayes: aggregating one-dependence estimators. *Mach Learn* 58(1):5–24
20. Zaidi NA, Cerquides J, Carman MJ, Webb GI (2013) Alleviating naïve Bayes attribute independence assumption by attribute weighting. *J Mach Learn Res* 14(1):1947–1988
21. Zheng F, Webb GI (2006) Efficient lazy elimination for averaged one-dependence estimators. In: *Proceedings of the 23rd international conference on machine learning. ACM*, pp 1113–1120



Zhong-Liang Xiang earned a Ph.D. in computer science from Dongseo University in 2015. He received a science master degree in computer science at Ocean University of China in 2010 and a Bachelor of Science (BS) degree in computer science at Mudanjiang Normal University in 2003. His research interests include machine learning, especially in deep belief network and nonparametric Bayesian models.



Dae-Ki Kang is a professor at Dongseo University in South Korea. He was a senior member of engineering staff at the attached Institute of Electronics and Telecommunications Research Institute in South Korea. He earned a Ph.D. in computer science from Iowa State University in 2006. His research interests include intrusion detection, security informatics, ontology learning, and relational learning. Prior to joining Iowa State, he worked at a Bay-area startup company and at the Electronics and Telecommunication Research Institute in South Korea. He received a science master degree in computer science at Sogang University in 1994 and a bachelor of engineering (BE) degree in computer science and engineering at Hanyang University in 1992.