Research Article

Liangjun Yu*, Di Wang, Xian Zhou, and Xiaomin Wu

Hybrid modeling of structure extension and instance weighting for naive Bayes

https://doi.org/10.1515/jisys-2024-0400
received September 20, 2024; accepted December 09, 2024

Abstract: Due to robustness and efficiency, naive Bayes (NB) remains among the top ten data mining algorithms. However, the required conditional independence assumption more or less limits its classification performance. Of numerous approaches to improving NB, structure extension and instance weighting have both achieved remarkable improvements. To make full use of their complementary and consensus advantages, this article proposes a hybrid modeling approach to combining structure extension with instance weighting. We call the resulting model instance weighted averaged one-dependence estimators (IWAODE). In IWAODE, the dependencies among attributes are modeled by an ensemble of one-dependence estimators, and the corresponding probabilities are estimated from attribute value frequency-weighted training instances. The classification performance of IWAODE is experimentally validated on a large number of datasets.

Keywords: naive Bayes, structure extension, instance weighting, hybrid modeling

1 Introduction

In the field of machine learning, numerous approaches exist for addressing classification tasks [1–3]. Among them, Bayesian networks (BNs) not only have a strict probability basis, but also combine the intuitiveness of graph theory, and plays an increasingly crucial role in the application of machine learning. In recent years, BNs have been widely used in text classification [4], fault diagnosis [5], risk assessment [6], medical system [7], and other application fields.

Since learning the optimal BN classifier is considered to be an NP-difficult problem [8], more and more scholars focus on the improvement of naive Bayes (NB) network classifier. The NB classifier exhibits a simplistic architecture. Assuming there are four attributes, namely A_1, A_2, A_3, A_4, with class variable denoted as C, the structure of NB can be depicted in Figure 1. NB assumes that each attribute is independent of all the other attributes.

For a test instance $x = \langle a_1, a_2, \ldots, a_m\rangle$, the classification formula for the NB classifier is represented by the following equation:

*** Corresponding author: Liangjun Yu,** College of Computer, Hubei University of Education, Wuhan, Hubei, 430205, China; Hubei Co-Innovation Center of Basic Education Information Technology Services, Hubei University of Education, Wuhan, Hubei, 430205, China, e-mail: yuliangjun@hue.edu.cn
Di Wang: College of Computer, Hubei University of Education, Wuhan, Hubei, 430205, China; School of Computer and Information Engineering, Hubei Normal University, Huangshi, Hubei, 435002, China, e-mail: wangdi@hue.edu.cn
Xian Zhou: College of Computer, Hubei University of Education, Wuhan, Hubei, 430205, China; School of Computer and Information Engineering, Hubei Normal University, Huangshi, Hubei, 435002, China, e-mail: zhouxian@hue.edu.cn
Xiaomin Wu: Electric Power Research Institute, State Grid Hubei Electric Power Co., Ltd., Wuhan, Hubei, 430077, China, e-mail: wxm@cug.edu.cn

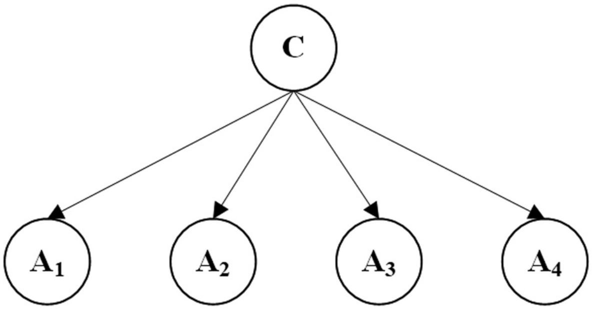

Figure 1: Structure of NB. Source: Created by the authors.

$$c(x) = \arg\max_{c\in C} P(c)\prod_{i=1}^{m} P(a_i|c), \tag{1}$$

where m is the number of attributes, a_i is the value of the ith attribute. C denotes the set of all potential class labels and $c(x)$ represents the predicted class label of x. The calculation formulas for the prior probability $P(c)$ and the conditional probability $P(a_i|c)$ are given by equations (2) and (3):

$$P(c) = \frac{1 + \sum_{t=1}^{n}\delta(c_t, c)}{q + n}, \tag{2}$$

$$P(a_i|c) = \frac{\sum_{t=1}^{n}\delta(a_{ti}, a_i)\delta(c_t, c) + 1}{\sum_{t=1}^{n}\delta(c_t, c) + n_i}, \tag{3}$$

where n is the number of training instances, q is the number of classes, c_t is the class label of the tth training instance, a_{ti} is the ith attribute value of the tth instance, n_i is the number of values for the ith attribute, and $\delta(\bullet)$ is an indicator function that equals 1 when two values are equal and 0 otherwise.

The NB classifier believes that attributes are independent of each other, but its attribute conditional independence assumption is difficult to hold in real life. Many researchers have conducted extensive research on enhancing NB from five different directions, such as structure extension [9–11], attribute weighting [12–15], attribute selection [16–20], instance weighting [21–23], and instance selection [24–26]. For example, SNB [19] is an efficient selective NB algorithm, which adopts only some of the attributes to construct selective NB, models and thus maintains the simplicity and efficiency of the algorithm. The naive Bayes enrichment method (NBEM) [20] is based on automated feature selection using threshold learning and the division of a dataset into sub-datasets according to the feature type. Numerous research studies have demonstrated that NB's classification effectiveness can be significantly enhanced through these five directions. Through extensive literature research, it has been observed that current approaches to enhancing NB classifiers predominantly concentrate on one of the five directions.

Structure extension, as the key direction of improving NB classifier, has received considerable scholarly attention [27,28]. Structural extension can effectively address the inherent constraint of NB by introducing directed edges to explicitly capture the interdependencies among attributes, thereby enhancing its modeling capability. The approach of averaged one-dependence estimator (AODE) involves selecting a specific class of one-dependence estimators and consolidating the predictions of all suitable classifiers, wherein a single attribute acts as the parent to all other attributes [29]. The improved model called hidden multinomial NB, which synthesizes the influence of other attributes by creating a hidden parent node for each attribute, is designed, which avoids the structure learning process with high computational complexity [30]. Attribute value-weighted average of one-dependence estimators assign discriminative weights to different one-dependence estimators by computing the correlation between the root attribute value and the class [31]. The BN model, based on the extension of NB, can clearly express the dependency relationship between attributes. By using directed arcs, the joint probability distribution can be explicitly expressed. Based on the above reasons, calculations involving conditional probability and prior probability are more accurate.

Instance weighting serves as an effective method to enhance NB by constructing an NB classifier on an instance weighted dataset. This process involves calculating the discriminative weight of each instance based on its distribution. Instance weighting is a highly effective direction for mitigating the primary limitation of NB [32]. The classification performance is enhanced through the utilization of instance weighting, whereby an NB classifier is trained on a set of training instances that have been assigned appropriate weights. The attribute value frequency weighted naive Bayes (AVFWNB) is a straightforward and effective eager learning approach that places emphasis on the frequency of each attribute value in order to determine the weight assigned to each instance [23]. An innovative and enhanced approach, known as attribute and instance weighted naive Bayes (AIWNB), integrates attribute weighting and instance weighting to effectively enhance its classification performance [28]. It acknowledges that certain training instances possess greater reliability and thus should exert a more substantial influence on the final model, in contrast to less reliable instances. Consequently, instance weighting assigns distinct weights to individual instances, which are subsequently integrated into the computation of prior and conditional probabilities with the aim of enhancing the accuracy of probability estimation.

To the extent of our understanding, however, the existing structure extension approaches regard each instance as equally important during modeling and do not pay attention to different influences of different instances on probability estimations. Furthermore, the existing instance weighting approaches impose a constraint on the model, restricting it to NB in the context of instance weighting modeling. The existing instance weighting approaches have hardly been studied on models with structure extension. It would be intriguing to investigate whether a hybrid model that combines structure extension with instance weighting can yield superior classification performance. The resulting model, which integrates instance weighting with a structural extension model, not only captures attribute dependencies through directed edges but also accounts for the varying impact of individual instances on classification performance.

In this study, we propose a hybrid model called instance weighted averaged one-dependence estimators (IWAODE), which integrates structure extension with instance weighting. Our IWAODE approach combines instance weighting with the AODE into one uniform framework. In contrast to the AODE, our improved IWAODE model is built on the instance weighted dataset. Instance weights are incorporated into reflecting mixture dependencies of both attributes and instances. Meanwhile, instance weights are calculated by the attribute value frequency-based instance weighted filter. Each instance weight is incorporated into probability estimates and the classification formula in IWAODE. We have concluded the experiments that involved comparing IWAODE against NB, AODE, and other advanced competitors. Extensive experiments demonstrate that our IWAODE approach surpasses its competitors in terms of classification performance. To sum up, the main contributions of this study are as follows:

(1) We find that the existing structure extension approaches regard each instance as equally important during modeling and the existing instance weighting approaches have hardly been studied on models with structure extension.
(2) We argue that a hybrid model that combines structure extension with instance weighting can yield superior classification performance.
(3) We propose a hybrid model called IWAODE, which not only captures attribute dependencies through directed edges but also accounts for the varying impact of individual instances.
(4) We conduct comprehensive experiments on 36 widely used datasets to evaluate the effectiveness of our proposed IWAODE. The experimental results demonstrate that IWAODE surpasses NB, AODE, and other advanced competitors.

The structure of the article is as follows. Section 2 reviews the related work. Section 3 proposes our IWAODE. Section 4 presents experiments and results. Section 5 outlines our conclusions and future works.

2 Related work

2.1 Structure extension

Structure extension improves the structural integrity of NB by introducing directed edges connecting attributes. In the extended structure model of NB, the classification performance is significantly improved by incorporating directed edges among attributes, thereby alleviating the assumption of conditional independence of NB. To address the primary limitation of NB, it is imperative to employ appropriate structures that can effectively mitigate the assumption of conditional independence [32].

Various approaches have been proposed to mitigate the assumption of independence in NB by extending its structure. The tree-augmented naive Bayes (TAN) assumes that the structure of the BN, which is composed of attribute variables, forms a tree [9,33,34]. With the exception of the attribute represented by the root node, all other attributes in the tree have a single parent node originating from another attribute. The hidden naive Bayes (HNB) is an enhanced approach that effectively integrates hidden dependencies among attributes [35]. Each attribute is accompanied by a hidden parent node. The hidden parent node incorporates the collective impact of all other attributes on this specific attribute.

A classical structure extension approach called AODE is proposed to enhance the structure of the NB network by considering each attribute as a parent node for other attributes, thereby relaxing the independence assumption of NB [29]. AODE learns a special tree expansion topology structure for each attribute node. As a classic model in structural extension, AODE avoids the process of learning topology structure and considerably improves the classification performance of NB. Given the assumption of four attributes, the AODE model encompasses four distinct topological configurations, as illustrated in Figure 2.

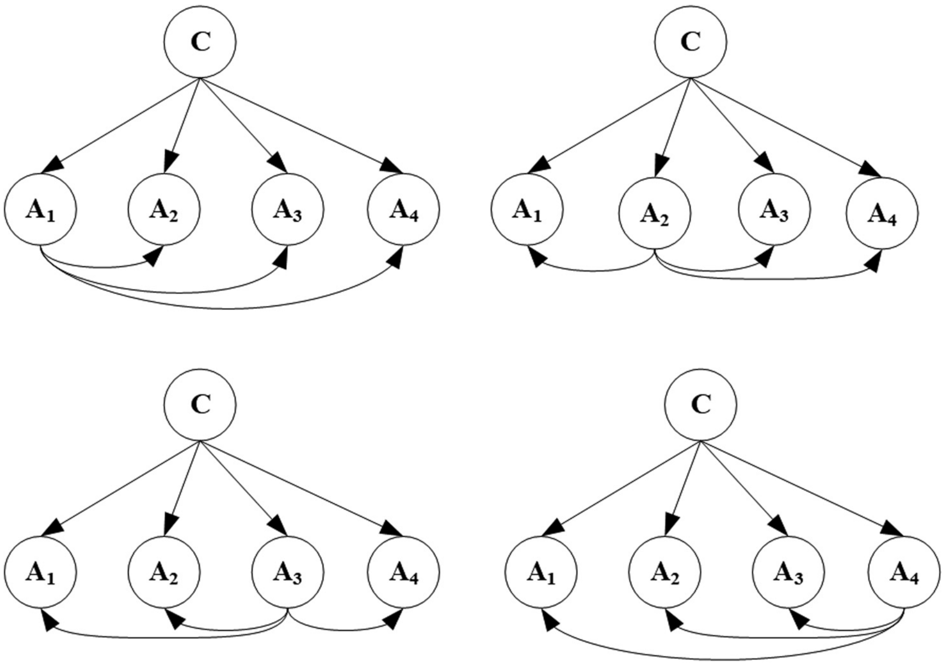

Figure 2: Structure of AODE. Source: Created by the authors.

Given a test instance x, AODE classifies it by equation (4):

$$\mathrm{c}(x) = \arg\max_{c \in C} \frac{\sum_{i=1 \Lambda F(a_i) \geq 30}^{m} P(a_i, c) \prod_{j=1 \Lambda \mathrm{j} \neq i}^{m} P(a_j | a_i, c)}{numParent}. \tag{4}$$

The frequency of occurrence, denoted as $F(a_i)$, quantifies the number of instances where attribute value a_i appears. To guarantee a sufficient number of instances in the training set for the attribute parent node to adopt this value, it is imperative that the count exceeds 30. This criterion ensures an adequate representation for statistical analysis while upholding the integrity. *numParent* represents the count of attribute nodes that meet the aforementioned criterion. The formulas for $P(a_i, c)$ and $P(a_j|a_i, c)$ are given by equations (5) and (6):

$$P(a_i, c) = \frac{\sum_{t=1}^{n}\delta(a_{ti}, a_i)\delta(c_t, c) + 1}{n + n_i{}^*q}, \tag{5}$$

$$P(a_j|a_i, c) = \frac{\sum_{t=1}^{n}\delta(a_{tj}, a_j)\delta(a_{ti}, a_i)\delta(c_t, c) + 1}{\sum_{t=1}^{n}\delta(a_{ti}, a_i)\delta(c_t, c) + n_j}, \tag{6}$$

where n_i represents the number of values that the parent attribute A_i can take, n_j represents the number of values that the child attribute A_j can take.

The efficacy of structure extension to enhance the classification performance of NB has been extensively validated by a multitude of studies. Adding directed edges to represent dependencies between attributes effectively strikes a well-balanced compromise between approximating ground-truth dependencies and ensuring the effectiveness of probability estimation. However, the construction with structure extension did not consider the varying impacts of different instances on classification performance. The calculation of the classification formula and probabilities assumes equal weights for all instances, disregarding the practical variation in their influence.

2.2 Instance weighting

Instance weighting is to assign distinct weights to individual instances during the training phase and subsequently construct the NB classifier using the instance weighted dataset. The detailed classification equation is

$$c(x) = \arg\max_{c\in C} P(c)\prod_{i=1}^{m}P(a_i|c). \tag{7}$$

The appearances of equations (7) and (1) are indistinguishable, yet they convey distinct connotations. $P(c)$ and $P(a_i|c)$ in equation (1) is performed on the original training dataset, whereas the computation of the aforementioned probabilities in equation (7) is conducted on the instance weighted training dataset. $P(c)$ and $P(a_i|c)$ in the instance weighted NB are given by equations (8) and (9):

$$P(c) = \frac{\sum_{t=1}^{n}w_t\delta(c_t, c) + 1}{\sum_{t=1}^{n}w_t + q}, \tag{8}$$

$$P(a_i|c) = \frac{\sum_{t=1}^{n}w_t\delta(a_{ti}, a_i)\delta(c_t, c) + 1}{\sum_{t=1}^{n}w_t\delta(c_t, c) + n_i}, \tag{9}$$

where $w_1, w_2, \ldots, w_n$ represent the different weights assigned to n distinct training instances. When all instances are assigned equal weights of 1, the instance weighted NB classifier simplifies to a standard NB classifier.

Instance weighting methods can be broadly categorized into two main groups: eager learning and lazy learning. In eager learning, instance weights are calculated during the training phase based on general characteristics of instances as a preprocessing step before the classification phase. On the other hand, lazy learning optimizes instance weights at the classification phase by employing search algorithms. Although eager learning typically has a quicker computation of instance weights than lazy learning, the latter exhibits superior classification performance.

The key to learning an instance weighted NB classifier lies in devising an efficient approach for weighting instances [36]. The most effective approach for instance weighting involves assigning weights to training instances based on the similarity between training and test instances. Training instances with higher weights exert a greater influence on the classification of a test instance, particularly those that are in close proximity.

The AVFWNB is a simple yet effective eager learning method that prioritizes the frequency of each attribute value to calculate weight to each instance [23]. The innovative approach, called AIWNB, amalgamates attribute weighting and instance weighting to significantly improve its classification performance [28]. To obtain instance weights, both eager and lazy approaches are adopted, resulting in two distinct versions. Discriminatively weighted NB employs the estimated loss of conditional probability to calculate instance weights, resulting in an eager learning approach that achieves outstanding classification results in terms of both accuracy and ranking [21]. The locally weighted NB (LWNB) employs instance weighting techniques, where each training instance is assigned a weight based on its distance to the test instance by selecting the k-nearest neighbors of the test instance [37]. The instance clone naive Bayes (ICNB) [38] is a lazy learning approach that clones training instances by evaluating the similarity between training and test instances, thereby constructing an extended dataset for developing an NB classifier to predict results for test instances.

The effectiveness of the instance weighting in enhancing the classification performance of NB has been extensively demonstrated by numerous studies. The existing instance weighting method of the BN, however, is primarily an enhancement of the NB. It overlooks attribute dependencies and solely focuses on the varying impacts of instances on classification performance.

3 IWAODEs

The research findings suggest that structure extension and instance weighting have the potential to enhance the classification performance of NB. Significant progress has been achieved by employing structure extension or instance weighting. Instance weighting involves assigning different weights to each instance, which are then incorporated into probability calculations in order to improve the classification performance. Structure extension enhances the structural integrity of NB by incorporating directed edges that establish connections between attributes, while treating each instance as equally important. Currently, there is limited research on the combination of structure extension and instance weighting. In order to fully exploit their synergistic benefits, our study aims to investigate whether better classification results can be achieved by combining structure extension with instance weighting in a new hybrid model.

This article proposed the IWAODE approach, by introducing a hybrid modeling that combines structure extension with instance weighting. The resulting hybrid model can not only capture attribute dependencies through directed edges but also account for the varying impact of individual instances on classification performance. The dependencies among attributes are modeled by an ensemble of one-dependence estimators. In order to maintain the simplicity, our IWAODE approach adopts an eager learning based on attribute value frequencies to calculate instance weights. It calculates distinct weights for each instance and incorporates these instance weights into model construction using a hybrid approach. Then, the corresponding probabilities are estimated using instance weighted training instances. In the subsequent subsection, we provide a comprehensive explanation of the IWAODE approach.

3.1 Modeling network structure

The AODE model is the classic model of the structure-extended NB, which significantly enhances the classification efficacy of the NB classifier [29,39]. AODE enhances the structure of the NB network by considering each attribute as a parent node for other attributes. However, the existing AODE assumes equal importance for each instance when computing probability estimates, which may not always hold true due to potential variations in contributions among different instances.

In order to calculate prior probabilities and conditional probabilities more accurately, the advantages of integrating instance weighting should be fully considered by taking into account in the structure-extended NB. Instance weighting can be employed to enhance the initial instance weights as a preliminary procedure for structure extension, thereby optimizing the overall process. Before constructing a structurally extended NB model, diverse weights are calculated for individual instances. Subsequently, these instance weights were integrated into the model of structure extension. In this study, we modify the AODE model into the IWAODE model. This enhanced model not only captures attribute dependencies through directed edges but also takes into account the varying impact of individual instances. The structure of IWAODE is visually illustrated in Figure 3. $w_1, w_2, \ldots, w_n$ represent the different weights of n different training instances. The improved model incorporates the integration of instance weights into the model construction process.

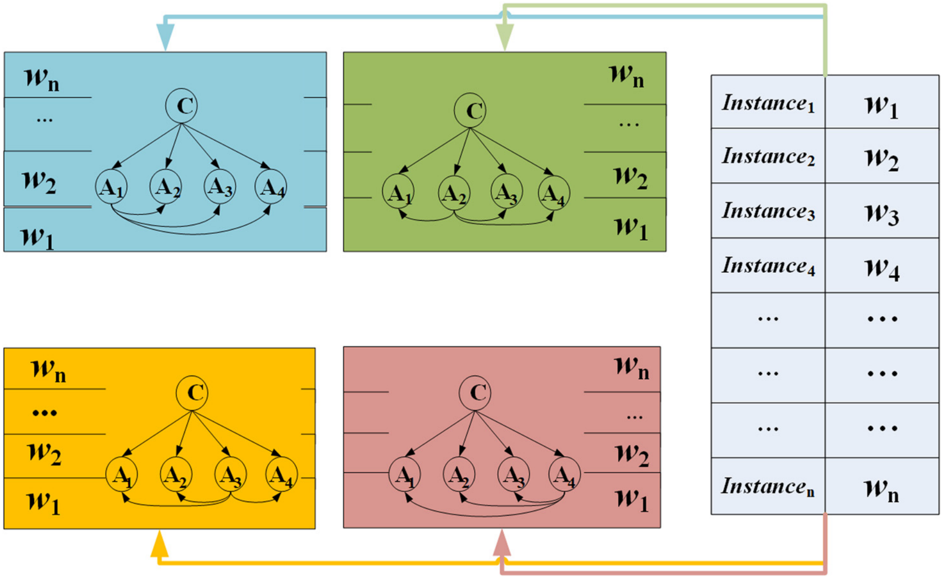

Figure 3: Structure of IWAODE. Source: Created by the authors.

Before the modeling process, the primary objective is to initially compute instance weights for each training instance. Subsequently, these instance weights are incorporated into multiple one-dependence estimators to enhance the accuracy of conditional probabilities and prior probabilities. It iterates over each attribute node, constructing multiple one-dependence estimators. Each attribute node is regarded as the parent node for all other attribute nodes. The dependency relationship between attributes is clearly expressed through directed edges. The improved IWAODE model can not only capture dependencies from all other attributes but also effectively represent varying contributions of different instances.

The classification equation utilized by the IWAODE model is represented as equation (10):

$$\mathrm{c}(x) = \arg\max_{c \in C} \frac{\sum_{i=1}^{m} P(a_i, c) \prod_{j=1 \wedge j \neq i}^{m} P(a_j | a_i, c)}{m}. \tag{10}$$

In this study, our IWAODE model embeds different instance weights. the prior probability $P(a_i, c)$ and conditional probability $P(a_j | a_i, c)$ are computed based on the instance weighted training dataset. So, we redefine both the prior probability and conditional probability. In equation (10), the prior probability $P(a_i, c)$ and conditional probability $P(a_j | a_i, c)$ are redefined as equations (11) and (12):

$$P(a_i, c) = \frac{\sum_{t=1}^{n} w_t \delta(a_{ti}, a_i) \delta(c_t, c) + 1}{\sum_{t=1}^{n} w_t + n_i {*} q}, \tag{11}$$

$$P(a_j | a_i, c) = \frac{\sum_{t=1}^{n} w_t \delta(a_{ti}, a_i) \delta(a_{tj}, a_j) \delta(c_t, c) + 1}{\sum_{t=1}^{n} w_t \delta(a_{ti}, a_i) \delta(c_t, c) + n_j}, \tag{12}$$

where n_i represents the number of values that the parent attribute A_i can take, n_j represents the number of values that the child attribute A_j can take, and $w_1, w_2, \ldots, w_n$ represent the distinct weights assigned to n diverse training instances. The redefined prior probability $P(a_i, c)$ and conditional probability $P(a_j|a_i, c)$ will be incorporated into the classification equation (10) to enhance the classification performance.

It is worthwhile to study how to calculate instance weights into the improved model, with the aim of enhancing its classification performance. Considering different contributions for individual instances becomes an essential factor. However, the calculation of different instance weights is a crucial matter that will be addressed in the subsequent subsection.

3.2 Modeling instance weights

In order to calculate prior probabilities and conditional probabilities more accurately, the advantages of integrating instance weighting should be fully considered by taking into account in the structure-extended NB. Instance weighting can be employed to enhance the initial instance weights as a preliminary procedure for structure extension, thereby optimizing the overall process.

It is crucial to acknowledge that certain instances in the training dataset contribute more significantly to classification and should have greater influence compared to less important ones. The calculation of instance weights constitutes the fundamental aspect of instance weighting. In order to maintain simplicity, our IWAODE approach employs an eager learning that utilizes attribute value frequencies for determining the weights of instances. It accurately and efficiently calculates the weight of each instance by leveraging vital information embedded in the frequency distribution of attribute values. The instance weights are found to have a positive correlation with both the attribute value number vector and the attribute value frequency vector.

The number of values varies for different attributes. The number of attribute values can indeed influence the attribute value frequencies to some extent. If an attribute has a larger number of possible values, the likelihood of a specific attribute value occurring is relatively smaller and *vice versa*. $\langle n_1, n_2, \ldots, n_m \rangle$ is used to indicate the numerical value of each attribute. The attribute value frequency denotes the ratio between the occurrence of attribute value and the overall number of instances [23]. To quantify the frequency of an attribute value, f_{ti} is employed to represent the frequency of the particular attribute value a_{ti}, a_{ti} denotes the ith attribute value of the tth instance. Equation (13) is formulated to denote the attribute value frequency:

$$f_{ti} = \frac{\sum_{r=1}^{n} \delta(a_{ri}, a_{ti})}{n}. \tag{13}$$

For the tth training instance, its attribute value frequency vector is denoted as $\langle f_{t1}, f_{t2}, \ldots, f_{tm} \rangle$. The greater the attribute value frequency, the stronger its influence on the instance. $\langle f_{t1}, f_{t2}, \ldots, f_{tm} \rangle$ can effectively reflect the significance of the instance. The tth instance weight w_t is determined by calculating the dot product between its attribute value frequency vector and its attribute value number vector:

$$w_t = \langle f_{t1}, f_{t2}, \ldots, f_{tm} \rangle \bullet \langle n_1, n_2, \ldots, n_m \rangle = \sum_{i=1}^{m} (f_{ti} * n_i). \tag{14}$$

IWAODE calculates the weight of each instance using Formula (14), and then embeds each instance weight into the IWAOE model. Each instance weight is embedded into the calculation of the redefined prior probability $P(a_i, c)$ and conditional probability $P(a_j|a_i, c)$, which leads to more accurate probabilities. Therefore, the classification equation (10) can not only capture attribute dependencies through directed edges but also account for the varying impact of individual instances on classification performance.

This study proposes the IWAODE model. Before the modeling process, it initially computes instance weights for each training instance. Subsequently, these instance weights are embedded into the computation of prior and conditional probabilities, and finally incorporates them into the classifier's classification formula. The detailed learning algorithm for our IWAODE model can be described as Algorithm 1.

Algorithm 1: Instance weighted averaged one-dependence estimator

```
Require: $TD$-a training dataset; a test instance $x$
Ensure: class label $c(x)$
for each training instance do
    Compute vector $\langle n_1, n_2, \ldots, n_m \rangle$
    Compute attribute value frequency vector by equation (13)
    Compute each instance weight by equation (14)
    Set each instance weighted
end for
The dataset $TD$ is transformed into the instance weighted dataset $TD^{IW}$
for each class label do
    Compute $P(a_i, c)$ by equation (11) from $TD^{IW}$
    foreach pair of attributes $(j \neq i)$ do
        Compute $P(a_j|a_i, c)$ by equation (12) from $TD^{IW}$
    end for
end for
Predict the class label $c(x)$ of $x$ by equation (10)
```

Based on the provided algorithm, the additional time needed for computing instance weights during training has a complexity of just $O(3nm)$ in comparison to AODE, where n signifies the count of training instances and m indicates the number of attributes. Therefore, the training time complexity of IWAODE can be expressed as $O(3nm + nm^2)$, which is relatively low. Furthermore, the time complexity for classifying with IWAODE is identical to that of AODE, which is $O(qn^2)$, where q is the number of classes. The overall time complexity of the IWAODE approach is $O(3nm + nm^2 + qn^2)$. In a word, the IWAODE model can be characterized as straightforward, highly efficient, and remarkably effective.

4 Experiments and results

To assess the performance of the IWAODE model, comparative experiments were conducted against state-of-the-art competitors, namely NB, AODE [29], AVFWNB [23], CFWNB [40], AIWNB [28], and HNB [35].

We conducted experiments on 36 UCI datasets [41] available on the official website[1] of the Waikato Environment for Knowledge Analysis (WEKA) platform [42], which represent a wide range of domains and data characteristics listed in Table 1. During our experiments, missing values were substituted with the modes and means of corresponding attribute values from the existing data. This step is crucial because most machine learning algorithms cannot handle missing values directly, and leaving them untreated could lead to biased or incomplete model training. Numeric attribute values were discretized by the Fayyad and Irani's MDL method [43]. This process converts continuous numeric attributes into a finite set of intervals or bins. If an attribute has as many unique values as there are instances, it does not provide any discriminative information and is therefore considered redundant. Consequently, we manually removed three such redundant attributes: "Hospital Number" in the dataset "colic.ORIG," "instance name" in the dataset "splice," and "animal" in the dataset "zoo." These preprocessing steps are essential for preparing the data for effective machine learning. They enhance the quality of datasets, making datasets more amenable to analysis and ensuring that the insights drawn from them are valid and actionable.

[1] https://waikato.github.io/weka-wiki/datasets/

Table 1: Description of datasets used in the experiments

Dataset	Instances	Attributes	Classes	Missing	Numeric
Anneal	898	39	6	Y	Y
Anneal.ORIG	898	39	6	Y	Y
Audiology	226	70	24	Y	N
Autos	205	26	7	Y	Y
Balance-scale	625	5	3	N	Y
Breast-cancer	286	10	2	Y	N
Breast-w	699	10	2	Y	N
Colic	368	23	2	Y	Y
Colic.ORIG	368	28	2	Y	Y
Credit-a	690	16	2	Y	Y
Credit-g	1,000	21	2	N	Y
Diabetes	768	9	2	N	Y
Glass	214	10	7	N	Y
Heart-c	303	14	5	Y	Y
Heart-h	294	14	5	Y	Y
Heart-statlog	270	14	2	N	Y
Hepatitis	155	20	2	Y	Y
Hypothyroid	3,772	30	4	Y	Y
Ionosphere	351	35	2	N	Y
Iris	150	5	3	N	Y
kr-vs-kp	3,196	37	2	N	N
Labor	57	17	2	Y	Y
Letter	20,000	17	26	N	Y
Lymph	148	19	4	N	Y
Mushroom	8,124	23	2	Y	N
Primary-tumor	339	18	21	Y	N
Segment	2,310	20	7	N	Y
Sick	3,772	30	2	Y	Y
Sonar	208	61	2	N	Y
Soybean	683	36	19	Y	N
Splice	3,190	62	3	N	N
Vehicle	846	19	4	N	Y
Vote	435	17	2	Y	N
Vowel	990	14	11	N	Y
Waveform-5000	5,000	41	3	N	Y
Zoo	101	18	7	N	Y

Table 2 presents the results of the comparisons in relation to the classification accuracy. The classification accuracy estimates were derived by averaging the results from ten distinct iterations using a stratified tenfold cross-validation. Meanwhile, two-tailed t-test with the $p = 0.05$ significance level [44] is used to compare the proposed IWAODE with its competitors. We utilize the symbol • to indicate that IWAODE exhibits a substantial enhancement compared to its competitors, while employing the symbol ∘ signifies a considerable decline in performance. The tables at the bottom provide a summary of the averages and $W/T/L$ values. The $W/T/L$ entries in the tables indicate that IWAODE outperforms its competitors on W datasets, has similar performance on T datasets, and performs worse on L datasets.

Table 3 presents a summary of the test results for IWAODE. Each entry $i(j)$ indicates the number of datasets where the model in the column outperforms the model in the corresponding row in terms of classification accuracy, with i representing total wins and j representing significant wins. Table 4 displays the ranking test outcomes for IWAODE. The first column denotes the difference between the total wins and losses compared to other models, which determines the ranking. The second and third columns show the total number of wins and losses, respectively. From these comparative results, we can draw that

Table 2: Classification accuracy comparisons for IWAODE versus NB, AODE, AVFWNB,CFWNB, AIWNB, and HNB

Dataset	IWAODE	NB	AODE	AVFWNB	CFWNB	AIWNB	HNB
Anneal	99.38 ± 0.80	96.13 ± 2.16 •	98.01 ± 1.39 •	98.62 ± 1.15 •	98.50 ± 1.29 •	98.94 ± 1.05	98.33 ± 1.22 •
Anneal.ORIG	93.85 ± 2.54	92.66 ± 2.72 •	93.35 ± 2.53	93.32 ± 2.65	94.60 ± 2.48	95.06 ± 2.23	95.29 ± 2.04
Audiology	80.18 ± 8.24	71.40 ± 6.37 •	71.66 ± 6.42 •	78.58 ± 8.44	74.22 ± 6.36 •	83.93 ± 7.00 ∘	69.04 ± 5.83 •
Autos	85.31 ± 7.96	72.30 ± 10.31 •	80.74 ± 8.68	77.27 ± 9.43 •	77.95 ± 8.95 •	78.04 ± 9.02 •	82.17 ± 8.60
Balance-scale	69.28 ± 3.85	71.08 ± 4.29 ∘	69.34 ± 3.82	71.10 ± 4.30 ∘	73.76 ± 4.15 ∘	73.75 ± 4.22 ∘	69.05 ± 3.75
Breast-cancer	70.89 ± 7.18	72.94 ± 7.71	72.53 ± 7.15	71.41 ± 7.98	72.46 ± 7.25	71.90 ± 7.55	73.09 ± 6.11
Breast-w	97.07 ± 1.88	97.25 ± 1.79	96.97 ± 1.87	97.48 ± 1.68	97.14 ± 1.81	97.17 ± 1.68	96.32 ± 2.01
Colic	81.42 ± 5.98	81.39 ± 5.74	82.64 ± 5.83	81.47 ± 5.86	83.34 ± 5.62	83.45 ± 5.45	82.09 ± 5.86
Colic.ORIG	75.58 ± 6.49	73.62 ± 6.83	74.62 ± 6.51	72.91 ± 6.34	73.70 ± 6.46	73.87 ± 6.40	74.06 ± 5.79
Credit-a	86.35 ± 3.60	86.25 ± 4.01	86.71 ± 3.82	86.23 ± 3.85	86.99 ± 3.81	87.03 ± 3.83	85.91 ± 3.70
Credit-g	76.32 ± 3.88	75.43 ± 3.84	76.50 ± 3.89	75.38 ± 3.90	75.70 ± 3.53	75.81 ± 3.60	76.12 ± 3.72
Diabetes	78.06 ± 4.41	77.85 ± 4.67	78.07 ± 4.56	77.89 ± 4.66	78.01 ± 4.89	77.87 ± 4.86	76.81 ± 4.11
Glass	77.96 ± 8.70	74.39 ± 7.95	76.08 ± 8.07	76.25 ± 8.07	73.37 ± 8.38	74.02 ± 8.41	77.80 ± 8.40
Heart-c	82.90 ± 6.25	83.60 ± 6.42	83.20 ± 6.20	83.04 ± 6.68	82.94 ± 6.57	82.71 ± 6.61	82.31 ± 6.81
Heart-h	85.18 ± 5.86	84.46 ± 5.92	84.43 ± 5.92	84.90 ± 5.68	83.82 ± 6.16	84.29 ± 5.85	84.87 ± 6.03
Heart-statlog	83.26 ± 6.44	83.74 ± 6.25	83.33 ± 6.61	83.78 ± 6.29	83.44 ± 6.69	83.22 ± 6.61	82.33 ± 6.55
Hepatitis	85.52 ± 9.01	84.22 ± 9.41	84.98 ± 9.26	85.38 ± 9.00	85.95 ± 9.25	85.75 ± 8.97	88.26 ± 7.28
Hypothyroid	99.10 ± 0.47	98.48 ± 0.59 •	98.76 ± 0.54 •	98.98 ± 0.48	98.56 ± 0.56 •	99.07 ± 0.48	98.95 ± 0.48
Ionosphere	93.88 ± 3.65	90.77 ± 4.76	92.79 ± 4.26	91.94 ± 4.09	91.82 ± 4.34	92.40 ± 4.13	91.82 ± 4.33
Iris	93.27 ± 5.65	94.47 ± 5.61	93.20 ± 5.76	94.40 ± 5.50	94.40 ± 5.50	94.40 ± 5.50	93.80 ± 5.86
kr-vs-kp	91.71 ± 1.53	87.79 ± 1.91 •	91.01 ± 1.67 •	88.18 ± 1.86 •	93.58 ± 1.32 ∘	93.73 ± 1.28 ∘	92.36 ± 1.30
Labor	94.33 ± 10.67	93.13 ± 10.56	94.70 ± 9.15	94.33 ± 10.13	92.10 ± 10.94	94.33 ± 9.30	94.87 ± 9.82
Letter	91.65 ± 0.63	74.00 ± 0.88 •	88.76 ± 0.70 •	75.07 ± 0.84 •	75.22 ± 0.83 •	75.56 ± 0.89 •	88.20 ± 0.66 •
Lymphography	87.52 ± 7.49	84.97 ± 8.30	86.98 ± 8.32	85.49 ± 7.83	84.81 ± 8.13	84.68 ± 7.99	85.84 ± 8.86
Mushroom	99.96 ± 0.06	95.52 ± 0.78 •	99.95 ± 0.07	99.12 ± 0.31 •	99.19 ± 0.32 •	99.53 ± 0.23 •	99.94 ± 0.10
Primary-tumor	46.94 ± 5.86	47.20 ± 6.02	47.67 ± 6.30	45.85 ± 6.53	47.20 ± 5.27	47.76 ± 5.25	47.66 ± 6.21
Segment	97.11 ± 1.05	91.71 ± 1.68 •	95.77 ± 1.23 •	93.69 ± 1.41 •	93.47 ± 1.46 •	94.16 ± 1.38 •	95.88 ± 1.19 •
Sick	97.38 ± 0.80	97.10 ± 0.84 •	97.39 ± 0.79	97.02 ± 0.86 •	97.36 ± 0.84	97.33 ± 0.85	97.56 ± 0.74
Sonar	86.07 ± 6.58	85.16 ± 7.52	86.60 ± 6.91	84.49 ± 7.79	82.56 ± 8.25 •	82.23 ± 8.65 •	84.63 ± 7.34
Soybean	94.71 ± 2.15	92.20 ± 3.23 •	93.28 ± 2.84 •	94.52 ± 2.36	93.66 ± 2.73	94.74 ± 2.19	93.88 ± 2.47
Splice	96.42 ± 0.96	95.42 ± 1.14 •	96.12 ± 1.00 •	95.61 ± 1.11 •	96.19 ± 0.99	96.21 ± 0.99	95.84 ± 1.10 •
Vehicle	73.42 ± 3.51	62.52 ± 3.81 •	72.31 ± 3.62	63.36 ± 3.87 •	62.91 ± 3.88 •	63.59 ± 3.92 •	72.37 ± 3.35
Vote	94.50 ± 3.15	90.21 ± 3.95 •	94.52 ± 3.19	90.25 ± 3.95 •	92.11 ± 3.74 •	92.18 ± 3.76 •	94.43 ± 3.18
Vowel	89.32 ± 3.05	65.23 ± 4.53 •	80.87 ± 3.82 •	67.46 ± 4.62 •	68.84 ± 4.30 •	69.98 ± 4.11 •	85.12 ± 3.65 •
Waveform-5000	86.04 ± 1.55	80.72 ± 1.50 •	86.03 ± 1.56	80.65 ± 1.46 •	83.11 ± 1.38 •	82.98 ± 1.37 •	86.21 ± 1.44
Zoo	96.25 ± 5.57	93.98 ± 7.14	94.66 ± 6.38	96.05 ± 5.60	95.96 ± 5.61	96.05 ± 5.60	97.73 ± 4.64
Average	86.61	83.31	85.68	84.21	84.41	84.94	85.86
W/T/L	—	16/19/1	9/27/0	12/23/1	12/22/2	9/24/3	6/30/0

• and ∘ denote statistically significant improvement or degradation, respectively.

Table 3: Summary test results on classification accuracy

Algorithm	IWAODE	NB	AODE	AVFWNB	CFWNB	AIWNB	HNB
IWAODE	—	7 (1)	13 (0)	7 (1)	12 (2)	12 (3)	11 (0)
NB	29 (16)	—	29 (13)	26 (11)	26 (12)	27 (15)	27 (16)
AODE	23 (9)	7 (1)	—	12 (3)	14 (2)	15 (7)	18 (3)
AVFWNB	29 (12)	10 (0)	24 (10)	—	19 (5)	26 (10)	23 (11)
CFWNB	24 (12)	10 (0)	22 (8)	16 (1)	—	25 (7)	23 (8)
AIWNB	24 (9)	9 (0)	21 (8)	8 (0)	10 (0)	—	20 (7)
HNB	25 (6)	9 (1)	18 (1)	13 (3)	13 (3)	16 (4)	—

Table 4: Ranking test results on classification accuracy

Algorithm	Wins losses	Wins	Losses
IWAODE	57	64	7
HNB	27	45	18
AIWNB	22	46	24
AODE	15	40	25
CFWNB	−12	24	36
AVFWNB	−29	19	48
NB	−80	3	83

(1) IWAODE achieves the highest classification accuracy (86.61%) compared to its competitors such as NB (83.31%), AODE (85.68%), AVFWNB (84.21%), CFWNB (84.41%), AIWNB (84.94%), and HNB (85.86%).

(2) IWAODE outperformed NB (16 wins and 1 loss), AODE (9 wins and 0 losses), AVFWNB (12 wins and 1 losses), CFWNB (12 wins and 2 losses), AIWNB (9 wins and 3 losses), and HNB (6 wins and 0 losses).

(3) The test results for summary and ranking indicate that IWAODE performs the best overall, with 64 wins and only 7 losses. When sorting in descending order across all datasets, the rankings are as follows: IWAODE is at the top, followed by HNB, AIWNB, AODE, CFWNB, AVFWNB, and NB.

Based on the detailed classification accuracy comparisons in Table 2, we conducted a comprehensive comparison of each pair of models using the KEEL software [45]. The results obtained from the Wilcoxon signed rank test [46,47] are provided in Tables 5 and 6. The Wilcoxon signed rank test is a non-parametric statistical hypothesis test used to compare two related samples, matched samples, or repeated measurements on a single sample to assess whether their population mean ranks differ. This statistical test evaluates the performance differences between two classification models for each dataset, disregarding symbols, and compares the rankings of positive R^+ and negative R^- differences [46,47]. Considering datasets with confidence levels $p = 0.05$ ($p = 0.1$) and with datasets where $N = 36$, two classifiers are regarded as "significantly different" if either R^+ or R^- is equal to or less than 208 (227). The confidence level for the lower diagonal is $\alpha = 0.05$, while the confidence level for the upper diagonal is $\alpha = 0.1$. ∘ signifies that the models in the presentation column enhance the models in the corresponding row, while • indicates that the approaches in the presentation row improve those in the corresponding column. According to Tables 5 and 6, we can see that IWAODE performed significantly better than NB, AODE, AVFWNB, CFWNB, AIWNB, and HNB at a confidence level of $\alpha = 0.1$.

Table 5: Ranks of the Wilcoxon test on classification accuracy

Algorithm	IWAODE	NB	AODE	AVFWNB	CFWNB	AIWNB	HNB
IWAODE	—	588.5	497.0	544.5	506.0	426.5	468.0
NB	77.5	—	73.5	141.0	117.0	96.0	129.0
AODE	169.0	592.5	—	483.5	440.0	400.0	323.0
AVFWNB	85.5	525.0	182.5	—	259.0	142.0	173.0
CFWNB	160.0	513.0	226.0	371.0	—	136.5	196.0
AIWNB	203.5	570.0	266.0	488.0	493.5	—	254.0
HNB	198.0	537.0	343.0	493.0	434.0	412.0	—

In our experiments, we have also evaluated the performance of IWAODE in terms of the root mean square error (RMSE) [48,49]. The RMSE quantifies the difference between the predicted and actual values. In terms of RMSE, a higher value indicates poorer performance. The corresponding comparison results are shown in Tables 7–11. Please be aware that the interpretation of the symbols presented in these tables differs from that

Table 6: Summary of the Wilcoxon test on classification accuracy

Algorithm	IWAODE	NB	AODE	AVFWNB	CFWNB	AIWNB	HNB
IWAODE	—	•	•	•	•	•	•
NB	∘	—	∘	∘	∘	∘	∘
AODE	∘	•	—	•	•		
AVFWNB	∘	•	∘	—		∘	∘
CFWNB	∘	•			—	∘	∘
AIWNB		•		•	•	—	
HNB	∘	•		•			—

Table 7: RMSE comparisons for IWAODE versus NB, AODE, AVFWNB,CFWNB, AIWNB, HNB

Dataset	IWAODE	NB	AODE	AVFWNB	CFWNB	AIWNB	HNB
Anneal	0.04 ± 0.03	0.09 ± 0.03 ∘	0.07 ± 0.02 ∘	0.06 ± 0.03 ∘	0.07 ± 0.02 ∘	0.05 ± 0.03 ∘	0.06 ± 0.02 ∘
Anneal.ORIG	0.12 ± 0.02	0.13 ± 0.02 ∘	0.13 ± 0.02 ∘	0.12 ± 0.02 ∘	0.12 ± 0.02	0.12 ± 0.02	0.11 ± 0.02
Audiology	0.11 ± 0.02	0.14 ± 0.01 ∘	0.14 ± 0.01 ∘	0.12 ± 0.02 ∘	0.12 ± 0.01	0.10 ± 0.02 •	0.14 ± 0.01 ∘
Autos	0.18 ± 0.06	0.25 ± 0.05 ∘	0.21 ± 0.05	0.23 ± 0.06 ∘	0.21 ± 0.05 ∘	0.21 ± 0.05 ∘	0.20 ± 0.05
Balance-scale	0.32 ± 0.02	0.33 ± 0.01 ∘	0.32 ± 0.02 ∘	0.33 ± 0.01 ∘	0.36 ± 0.01 ∘	0.36 ± 0.01 ∘	0.32 ± 0.02 •
Breast-cancer	0.45 ± 0.05	0.45 ± 0.06	0.44 ± 0.05 •	0.47 ± 0.06	0.44 ± 0.05	0.44 ± 0.05	0.44 ± 0.04
Breast-w	0.15 ± 0.06	0.15 ± 0.06	0.15 ± 0.05	0.14 ± 0.06	0.14 ± 0.06	0.14 ± 0.06	0.16 ± 0.04
Colic	0.38 ± 0.06	0.40 ± 0.06	0.37 ± 0.06	0.40 ± 0.06 ∘	0.35±0.06 •	0.35 ± 0.06 •	0.37 ± 0.06
Colic.ORIG	0.41 ± 0.05	0.41 ± 0.05	0.40 ± 0.05	0.44 ± 0.05 ∘	0.40 ± 0.04	0.41 ± 0.04	0.41 ± 0.04
Credit-a	0.32 ± 0.04	0.33 ± 0.04	0.32 ± 0.04	0.33 ± 0.04	0.31 ± 0.04	0.31 ± 0.04 •	0.32 ± 0.04
Credit-g	0.41 ± 0.02	0.41 ± 0.02	0.41 ± 0.02	0.41 ± 0.03	0.41 ± 0.02	0.41 ± 0.02	0.40 ± 0.02
Diabetes	0.39 ± 0.03	0.39 ± 0.03 ∘	0.39 ± 0.03	0.39 ± 0.03 ∘	0.39 ± 0.03	0.39 ± 0.03	0.39 ± 0.03
Glass	0.22 ± 0.03	0.23 ± 0.03	0.22 ± 0.03	0.23 ± 0.04	0.23 ± 0.02	0.23 ± 0.03 ∘	0.22 ± 0.03
Heart-c	0.22 ± 0.04	0.22 ± 0.05	0.22 ± 0.04	0.22 ± 0.05 ∘	0.21 ± 0.04	0.21 ± 0.04	0.21 ± 0.04
Heart-h	0.21 ± 0.04	0.22 ± 0.04 ∘	0.21 ± 0.04	0.21 ± 0.04	0.21 ± 0.03	0.21 ± 0.03	0.21 ± 0.03
Heart-statlog	0.34 ± 0.07	0.35 ± 0.07 ∘	0.34 ± 0.06	0.35 ± 0.07 ∘	0.33 ± 0.05	0.33 ± 0.05	0.34 ± 0.06
Hepatitis	0.31 ± 0.11	0.32 ± 0.12	0.31 ± 0.11	0.32 ± 0.12	0.30 ± 0.10	0.30 ± 0.10	0.29 ± 0.10
Hypothyroid	0.06 ± 0.01	0.07 ± 0.01 ∘	0.07 ± 0.01 ∘	0.06 ± 0.01 ∘	0.07 ± 0.01 ∘	0.06 ± 0.01 ∘	0.06 ± 0.01 ∘
Ionosphere	0.22 ± 0.09	0.28 ± 0.09	0.25 ± 0.09	0.26 ± 0.08	0.26 ± 0.08	0.25 ± 0.08	0.25 ± 0.08
Iris	0.13 ± 0.09	0.13 ± 0.10	0.13 ± 0.08	0.13 ± 0.10	0.13 ± 0.08	0.13 ± 0.08	0.15 ± 0.06 ∘
kr-vs-kp	0.26 ± 0.01	0.30 ± 0.02 ∘	0.27 ± 0.01 ∘	0.30 ± 0.02 ∘	0.28 ± 0.01 ∘	0.28 ± 0.01 ∘	0.25 ± 0.01 •
Labor	0.12 ± 0.15	0.15 ± 0.14	0.14 ± 0.13	0.12 ± 0.15	0.18 ± 0.13	0.16 ± 0.14	0.15 ± 0.11
Letter	0.07 ± 0.00	0.12 ± 0.00 ∘	0.08 ± 0.00 ∘	0.12 ± 0.00 ∘	0.11 ± 0.00 ∘	0.11 ± 0.00 ∘	0.08 ± 0.00 ∘
Lymphography	0.22 ± 0.06	0.23 ± 0.07	0.22 ± 0.06	0.24 ± 0.06	0.23 ± 0.06	0.24 ± 0.05	0.22 ± 0.06
Mushroom	0.01 ± 0.01	0.18 ± 0.02 ∘	0.01 ± 0.01 ∘	0.07 ± 0.01 ∘	0.08 ± 0.02 ∘	0.06 ± 0.01 ∘	0.02 ± 0.01 ∘
Primary-tumor	0.18 ± 0.01	0.18 ± 0.01 •	0.18±0.01 •	0.18 ± 0.01 ∘	0.18±0.01 •	0.18 ± 0.01 •	0.18 ± 0.01 •
Segment	0.08 ± 0.01	0.14 ± 0.01 ∘	0.10 ± 0.01 ∘	0.12 ± 0.01 ∘	0.12 ± 0.01 ∘	0.11 ± 0.01 ∘	0.10 ± 0.01 ∘
Sick	0.15 ± 0.02	0.16 ± 0.02 ∘	0.15 ± 0.02	0.16 ± 0.02 ∘	0.15±0.02 •	0.15 ± 0.02 •	0.14 ± 0.02 •
Sonar	0.32 ± 0.08	0.34 ± 0.09	0.32 ± 0.08	0.34 ± 0.09 ∘	0.34 ± 0.08	0.35 ± 0.08	0.31 ± 0.07
Soybean	0.07 ± 0.01	0.08 ± 0.02 ∘	0.07 ± 0.02 ∘	0.07 ± 0.02	0.07 ± 0.01 ∘	0.07 ± 0.01	0.07 ± 0.01
Splice	0.13 ± 0.02	0.15 ± 0.02 ∘	0.14 ± 0.02 ∘	0.15 ± 0.02 ∘	0.14 ± 0.02	0.14 ± 0.02	0.14 ± 0.02 ∘
Vehicle	0.29 ± 0.02	0.38 ± 0.02 ∘	0.30 ± 0.02 ∘	0.38 ± 0.02 ∘	0.36 ± 0.02 ∘	0.36 ± 0.02 ∘	0.29 ± 0.02
Vote	0.20 ± 0.07	0.29 ± 0.07 ∘	0.20 ± 0.07	0.29 ± 0.07 ∘	0.25 ± 0.07 ∘	0.25 ± 0.07 ∘	0.21 ± 0.07
Vowel	0.12 ± 0.01	0.21 ± 0.01 ∘	0.16 ± 0.01 ∘	0.20 ± 0.01 ∘	0.20 ± 0.01 ∘	0.20 ± 0.01 ∘	0.14 ± 0.01 ∘
Waveform-5000	0.25 ± 0.01	0.33 ± 0.01 ∘	0.25 ± 0.01	0.33 ± 0.01 ∘	0.28 ± 0.01 ∘	0.29 ± 0.01 ∘	0.25 ± 0.01
Zoo	0.06 ± 0.06	0.09 ± 0.05 ∘	0.08 ± 0.05 ∘	0.06 ± 0.07	0.09 ± 0.04 ∘	0.07 ± 0.06 ∘	0.07 ± 0.04
Average	0.21	0.24	0.22	0.23	0.23	0.22	0.21
W/T/L	—	21/14/1	14/20/2	23/13/0	14/19/3	14/17/5	9/23/4

∘ and • denote statistically significant improvement or degradation, respectively.

Table 8: Summary test results on RMSE

Algorithm	IWAODE	NB	AODE	AVFWNB	CFWNB	AIWNB	HNB
IWAODE	—	33 (21)	23 (14)	33 (23)	25 (14)	23 (14)	21 (9)
NB	3 (1)	—	2 (0)	13 (4)	6 (2)	6 (1)	3 (1)
AODE	13 (2)	34 (25)	—	26 (22)	21 (12)	20 (10)	14 (4)
AVFWNB	3 (0)	23 (12)	10 (2)	—	11 (4)	8 (2)	7 (1)
CFWNB	11 (3)	30 (17)	15 (3)	25 (15)	—	13 (2)	10 (3)
AIWNB	13 (5)	30 (18)	16 (4)	28 (20)	23 (12)	—	14 (3)
HNB	15 (4)	33 (19)	22 (6)	29 (16)	26 (15)	22 (11)	—

Table 9: Ranking test results on RMSE

Algorithm	Wins-Losses	Wins	Losses
IWAODE	80	95	15
HNB	50	71	21
AODE	46	75	29
AIWNB	22	62	40
CFWNB	16	43	59
AVFWNB	79	21	100
NB	103	9	112

Table 10: Ranks of the Wilcoxon test on RMSE

Algorithm	IWAODE	NB	AODE	AVFWNB	CFWNB	AIWNB	HNB
IWAODE	—	18.0	152.5	39.5	143.5	166.0	226.5
NB	648.0	—	612.0	464.5	553.0	595.0	593.0
AODE	477.5	18.0	—	129.0	199.0	246.5	420.5
AVFWNB	590.5	165.5	501.0	—	458.5	529.5	557.5
CFWNB	522.5	77.0	431.0	207.5	—	410.0	489.0
AIWNB	500.0	71.0	383.5	136.5	256.0	—	446.0
HNB	403.5	37.0	209.5	108.5	141.0	184.0	—

Table 11: Summary of the Wilcoxon test on RMSE

Algorithm	IWAODE	NB	AODE	AVFWNB	CFWNB	AIWNB	HNB
IWAODE	—	∘	∘	∘	∘	∘	
NB	•	—	•	•	•	•	•
AODE	•	∘	—	∘	∘		•
AVFWNB	•	∘	•	—	•	•	•
CFWNB	•	∘		∘	—		•
AIWNB	•	∘		∘		—	•
HNB		∘		∘	∘	∘	—

of Tables 2–6. In Table 7, we utilize the symbol ∘ to indicate that IWAODE exhibits a substantial enhancement compared to its competitors, while employing the symbol • signifies a considerable decline in performance. In Table 11, • signifies that the models in the presentation column enhance the models in the corresponding row, while ∘ indicates that the approaches in the presentation row improve those in the corresponding column. Table 7 shows the average RMSE results on 36 datasets. Tables 8 and 9 show the results for summary and ranking.

Table 10 and 11 show the experimental results of RMSE through the Wilcoxon signed rank test. According to the table, for the exact critical values in the Wilcoxon test at a confidence level of $p = 0.05$ ($p = 0.1$) and with datasets where $N = 36$, two algorithms are considered "significantly different" if the smaller of R^+ and R^- is equal to or less than 208 (227), and thus, we could reject the null hypothesis. ∘ indicates that the algorithm in the row improves the algorithm in the corresponding column, and • indicates that the algorithm in the column improves the algorithm in the corresponding row. Tables 10 and 11 show that IWAODE performed significantly better than NB, AODE, AVFWNB, CFWNB, and AIWNB at a confidence level of $\alpha = 0.05$.

From these comparison results on RMSE, we can draw that:

(1) IWAODE achieves the lowest RMSE (0.21) compared to its competitors such as NB (0.24), AODE (0.22), AVFWNB (0.23), CFWNB (0.23), AIWNB (0.22), and HNB (0.21).

(2) IWAODE outperformed NB (21 wins and 1 loss), AODE (14 wins and 2 losses), AVFWNB (23 wins and 0 losses), CFWNB (14 wins and 3 losses), AIWNB (14 wins and 5 losses), and HNB (9 wins and 4 losses).

(3) The test results for summary and ranking indicate that IWAODE performs the best overall. The rankings are as follows: IWAODE is at the top, followed by HNB, AODE, AIWNB, CFWNB, AVFWNB, and NB.

5 Conclusions and future works

This article proposes a hybrid modeling approach called IWAODE, which makes full use of the complementary and consensus advantages of structure extension and instance weighting to enhance the classification performance of NB. In IWAODE, the dependencies among attributes are modeled by an ensemble of one-dependence estimators. The utilization of directed arcs allows for the explicit expression of the joint probability distribution. IWAODE adopts an eager learning approach based on attribute value frequencies to calculate instance weights and incorporates these weights into model construction. In the initial phase of the modeling process, different weights are calculated for each training instance. Following this, these instance weights are integrated into the computation of both prior and conditional probabilities. Consequently, calculations pertaining to prior and conditional probabilities are rendered more precise. Extensive experimental results show that IWAODE obtains significant improvements in classification performance. Currently, there is limited research on the hybrid modeling of structure extension and instance weighting. The hybrid modeling approach called IWAODE proposed in this article demonstrates that the integration of structure extension and instance weighting can significantly enhance the classification performance.

Determining the optimal weights to accurately reflect the importance of different instances is crucial. Future improvements could involve utilizing more sophisticated algorithms for calculating instance weights, thereby further enhances the classification performance of IWAODE. Besides, adapting IWAODE for class-imbalance and cost-sensitive classification [50] is another research direction for our future work.

Funding information: This work was supported by Hubei Provincial Collaborative Innovation Center for Basic Education Information Technology Services No. OFHUE202312.

Author contributions: Liangjun Yu: conceptualization, methodology, formal analysis, programming, writing – review and editing. Di Wang: programming, formal analysis, data curation, writing – original draft preparation. Xian Zhou: formal analysis, data curation, writing – original draft preparation. Xiaomin Wu: formal analysis, data curation.

Conflict of interest: The authors declare no conflict of interest.

Data availability statement: The data analysed during the current study is available from https://waikato.github.io/weka-wiki/datasets/.

References

[1] Jiang L, Li C, Wang S, Zhang L. Deep feature weighting for naive Bayes and its application to text classification. Eng Appl Artif Intel. 2016;52:26–39. doi: 10.1016/J.ENGAPPAI.2016.02.002.

[2] Zhang L, Jiang L, Li C. A discriminative model selection approach and its application to text classification. Neural Comput Appl. 2019;31(4):1173–87. doi: 10.1007/S00521-017-3151-0.

[3] Chang V, Bailey J, Xu QA, Sun Z. Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms. Neural Comput Appl. 2023;35(22):16157–73. doi: 10.1007/S00521-022-07049-Z.

[4] Xue J, Liu K, Lu Z, Lu H. Analysis of Chinese comments on Douban based on naive Bayes. In: Proceedings of the 2nd International Conference on Big Data Technologies; 2019. p. 121–4. doi: 10.1145/3358528.3358570.

[5] Mack DL, Biswas G, Koutsoukos XD, Mylaraswamy D. Learning Bayesian network structures to augment aircraft diagnostic reference models. IEEE Trans Automat Sci Eng. 2016;14(1):358–69. doi: 10.1109/TASE.2016.2542186.

[6] Kelly DL, Smith CL. Bayesian inference in probabilistic risk assessment the current state of the art. Reliabil Eng Syst Safety. 2009;94(2):628–43. doi: 10.1016/J.RESS.2008.07.002.

[7] Bucci G, Sandrucci V, Vicario E, Ontologies and Bayesian networks in medical diagnosis. In: 2011 44th Hawaii International Conference on System Sciences. IEEE; 2011. p. 1–8. doi: 10.1109/HICSS.2011.333.

[8] Chickering DM. Learning Bayesian networks is NP-complete. In: Learning from Data - Fifth International Workshop on Artificial Intelligence and Statistics; 1995. p. 121–30. doi: 10.1007/978-1-4612-2404-4_12.

[9] Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers. Machine Learn. 1997;29(2–3):131–63. doi: 10.1023/A:1007465528199.

[10] Zheng X, Lin Z, Xu H, Chen C, Ye T. Efficient learning ensemble SuperParent-one-dependence estimator by maximizing conditional log likelihood. Expert Syst Appl. 2015;42(21):7732–45. doi: 10.1016/J.ESWA.2015.05.051.

[11] Xiang ZL, Kang DK. Attribute weighting for averaged one-dependence estimators. Appl Intel. 2017;46:616–29. doi: 10.1007/S10489-016-0854-3..

[12] Lee CH. A gradient approach for value weighted classification learning in naive Bayes. Knowledge-Based Syst. 2015;85:71–9. doi: 10.1016/J.KNOSYS.2015.04.020.

[13] Yu L, Jiang L, Wang D, Zhang L. Toward naive Bayes with attribute value weighting. Neural Comput Appl. 2019;31(10):5699–713. doi: 10.1007/S00521-018-3393-5.

[14] Jiang L, Zhang L, Yu L, Wang D. Class-specific attribute weighted naive Bayes. Pattern Recognit. 2019;88:321–30. doi: 10.1016/J.PATCOG.2018.11.032.

[15] Zhang H, Jiang L, Li C. Attribute augmented and weighted naive Bayes. Sci China Inform Sci. 2022;65(12):222101. doi: 10.1007/S11432-020-3277-0.

[16] Liu X, Lu R, Ma J, Chen L, Qin B. Privacy-preserving patient-centric clinical decision support system on naive Bayesian classification. IEEE J Biomed Health Inform. 2015;20(2):655–68. doi: 10.1109/JBHI.2015.2407157.

[17] Zhang L, Jiang L, Li C. A new feature selection approach to naive bayes text classifiers. Int J Pattern Recognit Artif Intell. 2016;30(2):1650003:1–17. doi: 10.1142/S0218001416500038.

[18] Lin J, Niu J, Li H. PCD: A privacy-preserving predictive clinical decision scheme with E-health big data based on RNN. In: 2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). IEEE; 2017. p. 808–13. doi: 10.1109/INFCOMW.2017.8116480.

[19] Chen S, Webb GI, Liu L, Ma X. A novel selective naïve Bayes algorithm. Knowl Based Syst. 2020;192:105361. doi: 10.1016/J.KNOSYS.2019.105361.

[20] Peretz O, Koren M, Koren O. Naive Bayes classifier - An ensemble procedure for recall and precision enrichment. Eng Appl Artif Intell. 2024;136:108972. doi: 10.1016/J.ENGAPPAI.2024.108972.

[21] Jiang L, Wang D, Cai Z. Discriminatively weighted naive Bayes and its application in text classification. Int J Artif Intell Tools. 2012;21(1):1250007. doi: 10.1142/S0218213011004770.

[22] Zhang Y, Wu J, Zhou C, Cai Z. Instance cloned extreme learning machine. Pattern Recognit. 2017;68:52–65. doi: 10.1016/J.PATCOG.2017.02.036.

[23] Xu W, Jiang L, Yu L. An attribute value frequency-based instance weighting filter for naive Bayes. J Experiment Theoretic Artif Intel. 2019;31(2):225–36. doi: 10.1080/0952813X.2018.1544284.

[24] Jiang L, Cai Z, Zhang H, Wang D. Naive Bayes text classifiers: a locally weighted learning approach. J Exp Theor Artif Intell. 2013;25(2):273–86. doi: 10.1080/0952813X.2012.721010.

[25] Wang S, Jiang L, Li C. Adapting naive Bayes tree for text classification. Knowl Inf Syst. 2015;44(1):77–89. doi: 10.1007/S10115-014-0746-Y.

[26] Bai Y, Wang H, Wu J, Zhang Y, Jiang J, Long G. Evolutionary lazy learning for Naive Bayes classification. In: 2016 International Joint Conference on Neural Networks (IJCNN). IEEE; 2016. p. 3124–9. doi: 10.1109/IJCNN.2016.7727597.

[27] Yu L, Gan S, Chen Y, He M. Correlation-based weight adjusted naive Bayes. IEEE Acc. 2020;8:51377–87. doi: 10.1109/ACCESS.2020.2973331.

[28] Zhang H, Jiang L, Yu L. Attribute and instance weighted naive Bayes. Pattern Recognit. 2021;111:107674. doi: 10.1016/J.PATCOG.2020.107674.

[29] Webb GI, Boughton JR, Wang Z. Not so naive Bayes: aggregating one-dependence estimators. Machine Learn. 2005;58:5–24. doi: 10.1007/S10994-005-4258-6.

[30] Gan S, Shao S, Chen L, Yu L, Jiang L. Adapting hidden naive Bayes for text classification. Mathematics. 2021;9:2378. doi: 10.3390/math9192378.

[31] Yu L, Jiang L, Wang D, Zhang L. Attribute value weighted average of one-dependence estimators. Entropy. 2017;19(9):501. doi: 10.3390/E19090501.

[32] Yu L, Gan S, Chen Y, Luo D. A novel hybrid approach: instance weighted hidden naive bayes. Mathematics. 2021;9:2982. doi: 10.3390/math9222982.

[33] Keogh E, Pazzani M. Learning augmented Bayesian classifiers: A comparison of distribution-based and classification-based approaches. In: Proceedings of the Seventh International Workshop on Artificial Intelligence and Statistics; 1999. p. 225–30.

[34] Qiu C, Jiang L, Li C. Not always simple classification: Learning SuperParent for class probability estimation. Expert Syst Appl. 2015;42(13):5433–40. doi: 10.1016/J.ESWA.2015.02.049.

[35] Jiang L, Zhang H, Cai Z. A novel Bayes model: Hidden naive Bayes. IEEE Trans Knowledge Data Eng. 2009;21(10):1361–71. doi: 10.1109/TKDE.2008.234.

[36] Zhang H, Jiang L, Yu L. Class-specific attribute value weighting for Naive Bayes. Inform Sci. 2020;508:260–74. doi: 10.1016/J.INS.2019.08.071.

[37] Frank E, Hall M, Pfahringer B. Locally weighted naive bayes. In: Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence. Morgan Kaufmann Publishers Inc.; 2002. p. 249–56. doi: 10.48550/arXiv.1212.2487.

[38] Jiang L, Wang D, Zhang H, Cai Z, Huang B. Using Instance cloning to Improve Naive Bayes for Ranking. Int J Pattern Recognit Artif Intell. 2008;22(6):1121–40. doi: 10.1142/S0218001408006703.

[39] Blanquero R, Carrizosa E, Ramírez-Cobo P, Sillero-Denamiel MR. Variable selection for Naïve Bayes classification. Comput Operat Res. 2021;135:105456. doi: 10.1016/J.COR.2021.105456.

[40] Jiang L, Zhang L, Li C, Wu J. A correlation-based feature weighting filter for naive Bayes. IEEE Trans Knowledge Data Eng. 2019;31(2):201–13. doi: 10.1109/TKDE.2018.2836440.

[41] Kelly M, Longjohn R, Nottingham K. The UCI Machine Learning Repository. https://archive.ics.uci.edu.

[42] Witten IH, Frank E, Hall MA. Data mining: practical machine learning tools and techniques, 3rd Edition. Morgan Kaufmann, Elsevier; 2011. doi: 10.1016/c2009-0-19715-5.

[43] Fayyad UM, Irani KB. Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning. In: Proceedings of the 13th International Joint Conference on Artificial Intelligence; 1993. p. 1022–9.

[44] Nadeau C, Bengio Y. Inference for the generalization error. Machine Learn. 2003;52(3):239–81. doi: 10.1023/A:1024068626366.

[45] Derrac J, Garcia S, Sanchez L, Herrera F. Keel data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. J Mult Valued Log Soft Comput. 2015;17:255–87. doi: 10.1016/j.jlap.2009.12.002.

[46] Windi WA, Taufiq M, Muhammad T. Implementasi Wilcoxon Signed Rank Test Untuk Mengukur Efektifitas Pemberian Video Tutorial Dan Ppt Untuk Mengukur Nilai Teori. Produktif: Jurnal Ilmiah Pendidikan Teknologi Informasi. 2021;5(1):405–10.

[47] Obulesu O, Kallam S, Dhiman G, Patan R, Kadiyala R, Raparthi Y, et al. Adaptive diagnosis of lung cancer by deep learning classification using Wilcoxon gain and generator. J Healthcare Eng. 2021;2021:5912051. doi: 10.1155/2021/5912051.

[48] Hodson TO. Root-mean-square error (RMSE) or mean absolute error (MAE): When to use them or not. Geoscientif Model Development. 2022;15(14):5481–7. doi: 10.5194/gmd-15-5481-2022.

[49] Chicco D, Warrens MJ, Jurman G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. PeerJ Comput Sci. 2021;7:e623. doi: 10.7717/PEERJ-CS.623.

[50] Zhang H, Jiang L, Li C. CS-ResNet: Cost-sensitive residual convolutional neural network for PCB cosmetic defect detection. Expert Syst Appl. 2021;185:115673. doi: 10.1016/J.ESWA.2021.115673.