mathematics

MDPI

Article

Attribute Selecting in Tree-Augmented Naive Bayes by Cross Validation Risk Minimization

Shenglei Chen [1,*], Zhonghui Zhang [2] and Linyuan Liu [1]

[1] Department of E-Commerce, Nanjing Audit University, Nanjing 211815, China; liulinyuang@nau.edu.cn
[2] School of Finance, Nanjing Audit University, Nanjing 211815, China; zhonghui@nau.edu.cn
* Correspondence: shenglei.chen@nau.edu.cn

Abstract: As an important improvement to naive Bayes, Tree-Augmented Naive Bayes (TAN) exhibits excellent classification performance and efficiency since it allows that every attribute depends on at most one other attribute in addition to the class variable. However, its performance might be lowered as some attributes might be redundant. In this paper, we propose an attribute Selective Tree-Augmented Naive Bayes (STAN) algorithm which builds a sequence of approximate models each involving only the top certain attributes and searches the model to minimize the cross validation risk. Five different approaches to ranking the attributes have been explored. As the models can be evaluated simultaneously in one pass learning through the data, it is efficient and can avoid local optima in the model space. The extensive experiments on 70 UCI data sets demonstrated that STAN achieves superior performance while maintaining the efficiency and simplicity.

Keywords: Tree-Augmented Naive Bayes; attribute selection; cross validation; mutual information

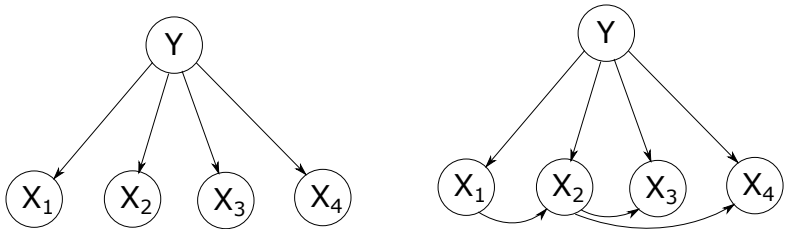
Citation: Chen, S.; Zhang, Z.; Liu, L. Attribute Selecting in Tree-Augmented Naive Bayes by Cross Validation Risk Minimization. *Mathematics* **2021**, *9*, 2564. https://doi.org/10.3390/math9202564

Academic Editor: Liangxiao Jiang

Received: 5 September 2021
Accepted: 9 October 2021
Published: 13 October 2021

1. Introduction

Naive Bayes (NB) [1,2] has attracted considerable attention due to its computational efficiency and competitive classification performance. Its efficiency originates in the independence assumption among the attributes given the class. Figure 1a describes an example of NB structure with 4 attributes, where $X_1, \ldots, X_4$ are the attribute variables and Y is the class variable. However, the independence assumption rarely holds in real world applications. Although it has been demonstrated that some violations of the independence assumption are not harmful to classification accuracy [3], it is clear that many are. Many efforts have been done to allow specific dependence between attributes while retaining naive Bayesisan classifiers' desirable simplicity and efficiency [4–6].



(**a**) NB (**b**) TAN

Figure 1. Structure of NB and TAN.

The Tree-Augmented Naive Bayes (TAN) [4] corresponds to the algorithm better improving the accuracy of naive Bayesian classifiers by alleviating its attribute independence assumption. TAN allows that every attribute depends on at most one attribute other than the class. So the dependencies among the attributes can be described by a tree structure, which can be found by a scoring measurement, called conditional mutual information. Figure 1b describes an example of TAN structure with 4 attributes, where X_2 depends only

on X_1, while X_3 and X_4 depend on X_2. Since TAN exploits the first-order dependencies among the attributes, the classification performance can be greatly improved compared to NB at the cost of a search of tree structure.

However, TAN demands all attributes to be connected to the class, so it exploits all the attributes regardless of whether it is redundant or not. As a result, there is an increasing body of work to improve TAN since TAN was proposed [7]. Jiang et al. [8] presented a Forest Augmented Naive Bayes (FAN) for better evaluating the ranking performance. Alhussan et al. [9] proposed a fine-tuning stage in a Bayesian Network (BN) learning algorithm to more accurately estimate the probability terms used by the BN. They apply the algorithm to fine-tune TAN and other models. Wang et al. [10] presented a kind of semi-lazy TAN Classifier, which builds a TAN identical to the original TAN at training time, but adjusts the dependence relations for a new test instance at classification time. Campos et al. [11] proposed an extended version of the well-known tree-augmented naive Bayes. The structure learning procedure explores a superset of the structures that are considered by TAN, yet achieves global optimality of the learning score function in a very efficient way. The procedure is enhanced with a new score function that only takes into account arcs that are relevant to predict the class, as well as an optimization over the equivalent sample size during learning. Jiang et al. [12] investigated the class probability estimation performance of TAN in terms of Conditional Log Likelihood (CLL). An improved algorithm was presented by choosing each attribute as the root of the tree and then averaging all the spanning TAN classifiers. Cerquides and Mántaras [13] introduced decomposable distributions over TANs. They proposed four classifiers based on decomposable distributions over TANs. These classifiers provide clearly significant improvements, specially when data is scarce. It could be found that these work focuses mainly on the TAN structure learning, few of them have tried to eliminate redundant attributes in the process of training.

Attribute selection in Bayesian network classifiers had been investigated. Zhang et al. [14] proposed a discriminative model selection approach which chooses different single models for different test instances and retains the interpretability of single models. Langley and Sage [15] proposed Selective Bayes Classifiers (SBC) by hill-climbing search of the optimal attribute subset. The same strategy had also been applied to Averaged-One Dependence Estimator (AODE) to find the optimal parent and child attribute set [16]. As this carries out a greedy search through the space of feature, it often falls into a local optimization. Furthermore, the evaluation of the successively added attributes is time-consuming. Recently, an attribute selection approach based on efficient attribute subsets construction and evaluation has been investigated in NB [17], k-dependence Bayesian classifier (KDB) [18], averaged n-dependence estimators (AnDE) [19,20]. However, the performance of TAN [4] with this attribute selection has not been explored.

This paper proposes an attribute Selective TAN (STAN) algorithm by cross validation risk minimization. The attribute subsets are first constructed very efficiently as the latter subset can be obtained only by adding one attribute to the former. Classification models based on these different subsets of attributes could then be searched by cross validation risk minimization in one pass learning of the training data. Five different approaches to ranking the attributes have been explored. Different from the traditional attribute selection based on hill climbing, the strategy in this paper is efficient and able to avoid local optima in the model space. The extensive experiments on 70 UCI data sets demonstrated that STAN, along with a certain attribute ranking approach, called Minimum Redundancy-Maximum Relevance (MRMR), achieves superior performance while maintaining the efficiency and simplicity. It provides consistently better predictions than regular TAN in a statistically way. The win/draw/loss result in terms of zero-one loss is 34/22/14, which means STAN with MRMR obtains lower zero-one loss than regular TAN on 34 data sets, the same zero-one loss as regular TAN on 22 data sets and greater zero-one loss than regular TAN on 14 data sets.

2. Preliminaries

2.1. Bayesian Network Classifiers

The classification problem can be described as a procedure that given a data set $\mathcal{D}$ and an unclassified observation $\mathbf{x}$ assigns a class to $\mathbf{x}$. Suppose we have N observations in $\mathcal{D}$. Each observation is a pair $(\mathbf{x}, y)$, consisting of an a-dimensional attribute vector $\mathbf{x} = [x_1, \ldots, x_a]^T$, and a target class y, draw from the underlying random variables $X = \{X_1, \ldots, X_a\}$ and Y.

Bayesian network classifier addresses this classification task by first modelling the joint distribution $P(y, \mathbf{x})$ by a certain Bayesian network $\mathcal{B}$, and then calculating the posterior distribution $P(y|\mathbf{x})$ by bayesian rule. A Bayesian network is characterised by a pair $\mathcal{B} = \langle \mathcal{G}, \Theta \rangle$. The first component, $\mathcal{G}$, is a directed acyclic graph. The nodes in $\mathcal{G}$ represent random variables, including attributes $X_1, \ldots, X_a$ and the class variable Y. The arcs in $\mathcal{G}$ represent directed dependencies between the nodes. If X_j is pointing directly to X_i via a directed edge (an arc), we say X_j is the parent of X_i, or X_i is the child of X_j. Different bayesian network classifiers assume different dependencies among the attributes, but all assume Y is the parent of all attributes and has no parents.

The second component of the pair, namely Θ, represents the set of parameters that quantifies the network. It contains a parameter $\theta_{x_i|y,\boldsymbol{\pi}_i}$ for each x_i of node X_i, each y of Y and each $\boldsymbol{\pi}_i$ of $\mathbf{\Pi}_i$, where $\mathbf{\Pi}_i$ is the set of parent nodes of X_i in network $\mathcal{G}$. $\theta_{x_i|y,\boldsymbol{\pi}_i} = P_{\mathcal{B}}(x_i|y, \boldsymbol{\pi}_i)$, abbreviated for $P_{\mathcal{B}}(X_i = x_i|Y = y, \mathbf{\Pi}_i = \boldsymbol{\pi}_i)$, represents the probability that variable X_i takes the value x_i given that Y takes the class y and $\mathbf{\Pi}_i$ takes the value $\boldsymbol{\pi}_i$. It is obvious that $\theta_{x_i|y,\boldsymbol{\pi}_i}$ is constrained by $\sum_{x_i \in X_i} \theta_{x_i|y,\boldsymbol{\pi}_i} = 1$.

When the data set $\mathcal{D}$ is given, the log-likelihood of the data given a specific network structure is maximized when $\theta_{x_i|y,\boldsymbol{\pi}_i}$ corresponds to empirical estimates of probabilities from the data, that is, $\theta_{x_i|y,\boldsymbol{\pi}_i} = P_{\mathcal{D}}(X_i = x_i|Y = y, \mathbf{\Pi}_i = \boldsymbol{\pi}_i)$ [21]. This produces the Maximum-Likelihood Estimation (MLE) of the parameters Θ.

A Bayesian network defines a unique joint probability distribution given by

$$P_{\mathcal{B}}(x_1, \ldots, x_a, y) = P_{\mathcal{B}}(y) \prod_{i=1}^{a} P_{\mathcal{B}}(x_i|y, \boldsymbol{\pi}_i) = P_{\mathcal{B}}(y) \prod_{i=1}^{a} \theta_{x_i|y,\boldsymbol{\pi}_i} \tag{1}$$

By Bayesian rule, the posterior distribution of a new unclassified example $\mathbf{x}$ can be calculated as follows,

$$P_{\mathcal{B}}(y|\mathbf{x}) = \frac{P_{\mathcal{B}}(\mathbf{x}, y)}{\sum_y P_{\mathcal{B}}(\mathbf{x}, y)} \tag{2}$$

So we can easily classify $\mathbf{x}$ into class $\arg\max_y(P_{\mathcal{B}}(y|\mathbf{x}))$.

2.2. Tree Augmented Naive Bayes

Although naive Bayes [1] performs surprisingly well on many data sets, its independence assumption among attributes rarely holds in real world. In order to relax this independence assumption, Friedman et al. [4] proposed to augment the naive Bayes structure with edges among the attributes, when needed, thus dispensing with its strong assumptions about independence. In order to learn the optimal set of augmenting edges in polynomial time, a tree restriction has been imposed on the form of the allowed interaction. So the resulting structure is called Tree-Augmented Naive Bayesian (TAN) network, in which the class variable has no parents and each attribute has as parents at most one other attribute in addition to the class variable. Thus, each attribute can have one augmenting edge pointing to it and all the augmented edges form a tree structure.

In order to learn a TAN structure such that the log likelihood is maximized, they proposed to use conditional mutual information between attributes given the class variable

as the weight of an edge in the graph. The conditional mutual information between attributes X and Z given the class variable Y is defined as

$$I(X;Z|Y) = \sum_{x,z,y} P(x,z,y) log \frac{P(x,z|y)}{P(x|y)P(z|y)}, \tag{3}$$

Roughly speaking, this function measures the information that Z provides about X when the value of Y is known.

The procedure to construct TAN structure consists of five main steps:

1. Compute $I(X_i;X_j|Y)$ between each pair of attributes, $i \neq j$, from the training data.
2. Build a complete undirected graph in which the nodes are the attributes $X_1,\ldots,X_a$. Annotate the weight of an edge connecting X_i to X_j by $I(X_i;X_j|Y)$.
3. Build a maximum weighted spanning tree.
4. Transform the resulting undirected tree to a directed one by choosing a root variable and setting the direction of all edges to be outward from it, thus getting the parent node Π_i of node X_i.
5. Construct a TAN model by adding a node labelled by Y and adding an arc from Y to each X_i.

Note that in TAN, vector $\mathbf{\Pi}_i$ has deteriorated to scalar Π_i as TAN allows only one parent for each attribute. So TAN model defines a unique joint probability distribution given by

$$P_{\text{TAN}}(\mathbf{x},y) = P_{\text{TAN}}(y)\prod_{i=1}^{a} P_{\text{TAN}}(x_i|y,\pi_i) \tag{4}$$

The structure and the parameters of TAN model can be learned in one pass learning through the data.

3. Attribute Selective Tree-Augmented Naive Bayes

3.1. Motivation

It could be found from Equation (4) that the joint probability $P_{\text{TAN}}(\mathbf{x},y)$ is estimated by the product of prior probability $P_{\text{TAN}}(y)$ and conditional probabilities $P_{\text{TAN}}(x_i|y,\pi_i)$. Considering only the top certain attributes will produce an approximation to $P_{\text{TAN}}(\mathbf{x},y)$. This implies that it is possible to build a sequence of alternative selective models such that the latter one is a trivial extension to the former. Different models that build upon one another in this way can be efficiently evaluated in a single set of computations. So in just one pass learning through the data, cross validation risks of different models can be obtained. Risk minimization will obtain the best model, which means also the optimal attribute subset to perform classification in the framework of TAN.

3.2. Building Model Sequence

When trying to search the best attribute subset, the size of the search space for a variables is 2^a. Instead of searching the whole space exhaustively, it is natural to impose some restrictions on the construction of the model space. As TAN compute the joint probability $P_{\text{TAN}}(\mathbf{x},y)$ by sequentially multiplying the conditional probability $P_{\text{TAN}}(x_i|y,\pi_i)$ to the prior $P_{\text{TAN}}(y)$, considering only the top s attributes results in an approximate model to $P_{\text{TAN}}(\mathbf{x},y)$, where $1 \leq s \leq a$. So the model space of attribute selective TAN would be

$$P_{\text{TAN}_s}(\mathbf{x},y) = P_{\text{TAN}}(y)\prod_{i=1}^{s} P_{\text{TAN}}(x_i|y,\pi_i) \tag{5}$$

By this means we can construct a model sequence of size a. These models can be evaluated efficiently in a single set of computations as the latter one is only a trivial extension to the former one. Although each model is only an approximation to TAN model,

regular TAN is also included in this model sequence. Consequently, this attribute selective model could be expected not to be worse than regular TAN.

3.3. Ranking the Attributes

Since the selective strategy is to consider only the top s attributes, the strategy relies on a ranking of the attributes. As the purpose of attribute selection is to eliminate those redundant attributes, we should prioritize those more informative attributes. We could rank the attributes based on the attributes' marginal relevance with respect to the class. Fortunately, the attribute ranking has been extensively investigated in feature selection area [22]. Here we adopt the most well known five strategies to measure the relevance between the attribute and the class.

1. Mutual Information (MI) (Mutual information measures the amount of information shared by two variables. This can also be interpreted as the amount of information that one variable provides about another) is an intuitive score since it is a measure of correlation between an attribute and the class variable. Before we present the definition of mutual information, we would first present the concepts of entropy and conditional entropy. The entropy of a random variable x is defined as

$$H(X) = -\sum_{x \in X} P(x) log_2 P(x) \tag{6}$$

The conditional entropy $H(X|Y)$ of X given Y is

$$H(X|Y) = -\sum_{y \in Y} P(y) \sum_{x \in X} P(x|y) log_2 P(x|y) \tag{7}$$

The mutual information between X and Y is defined as the difference between the entropy $H(X)$ and the conditional entropy $H(X|Y)$

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) \\ &= \sum_{y \in Y} \sum_{x \in X} P(x,y) log_2 \frac{P(x,y)}{P(x)P(y)}, \end{aligned} \tag{8}$$

This heuristic considers a score for each attribute independently of others.

2. Symmetrical Uncertainty (SU) (Symmetrical uncertainty is the normalized mutual information. The range of symmetrical uncertainty is $[0, 1]$, where the value 1 indicates that knowledge of the value of either one completely predicts the value of the other and the value 0 indicates that the two variables are independent) [23] can be interpreted as a sort of mutual information normalized to interval [0, 1]:

$$SU(X;Y) = 2\left(\frac{I(X;Y)}{H(X) + H(Y)}\right). \tag{9}$$

It is obvious that mutual information is biased in favor of attributes with more values and so large entropy. However, symmetrical uncertainty, which is normalized to the range $[0, 1]$, is an unbiased metric and ensures they are comparable and have the same effect. As a result, we can expect to obtain a more appropriate ranking of attributes based on symmetrical uncertainty.

3. Minimum Redundancy-Maximum Relevance (MRMR) criterion (MRMR, short for Minimum Reduncancy-Maximum Relevance, always tries to select the attribute which has the best trade off between the relevance to the class variable and the the averaged redundancy to the attributes already selected), which was proposed by Peng et al. [24], not only considers mutual information to ensure feature relevance, but introduces a penalty to enforce low correlations with features already selected. MRMR is very similar to Mutual Information Feature Selection (MIFS) [25], except that the latter

replace $\frac{1}{k}$ with a more general configurable parameter β, where k means the number of the attributes that have been selected so far, and is also the number of steps. Assume at step k, the attribute set selected so far is $\mathcal{A}_k^S$, while $\mathcal{A}_k^{-S} = \mathcal{A} \setminus \mathcal{A}_k^S$ is the set difference between the original set of inputs $\mathcal{A}$ and $\mathcal{A}_k^S$. The attribute returned by MRMR criterion at step $k+1$ is,

$$X_{k+1}^{\text{MRMR}} = \arg\max_{X \in \mathcal{A}_k^{-S}} \left\{ I(X;Y) - \frac{1}{k} \sum_{X' \in \mathcal{A}_k^S} I(X;X') \right\}. \tag{10}$$

At each step, this strategy selected the variable which has the best trade off between the relevance $I(X;Y)$ of X to the class Y and the averaged redundancy of X to the selected attributes $X' \in \mathcal{A}_k^S$.

4. Conditional Mutual Information Maximization (CMIM) (CMIM, short for Conditional Mutual Information Maximization, tries to select the attribute that maximizes the minimal mutual information with the class within the attributes already selected) proposes to select the feature whose minimal relevance conditioned to the selected attributes is maximal. This heuristic was proposed by Fleuret [26] and also later by Chen et al. [27] as direct rank (dRank). CMIM computes the mutual information of X and the class variable Y, conditioned on each attribute $X' \in \mathcal{A}_k^S$ previously selected. Then the minimal value is retained and the attribute that has a maximal minimal conditional relevance is selected.
   In formal notation, the variable returned at step $k+1$ according to the CMIM strategy is
$$X_{k+1}^{\text{CMIM}} = \arg\max_{X \in \mathcal{A}_k^{-S}} \left\{ \min_{X' \in \mathcal{A}_k^S} I(X;Y|X') \right\}. \tag{11}$$

5. Joint Mutual Information (JMI) (JMI, short for joint Mutual Information, tries to select the attribute which is complementary most with existing attributes), which was proposed by Yang and Moody [28] and also later by Meyer et al. [29], tries to select a candidate attribute if it is complementary with existing attributes. As a result, JMI focuses on increasing the complementary information between attributes. The variable returned by JMI criterion at step $k+1$ is
$$X_{k+1}^{\text{JMI}} = \arg\max_{X \in \mathcal{A}_k^{-S}} \left\{ \sum_{X' \in \mathcal{A}_k^S} I(X, X';Y) \right\}. \tag{12}$$

   The score in JMI is the information between the class variable and a *joint* random variable$\langle X, X' \rangle$, defined by pairing the candidate X with each attribute X' previously selected.

Note that for the first two scores, we can simply rank the attributes in the descending order of MI or SU scores. While for the last three methods, a forward selection search strategy is involved, which means we are selecting attributes *sequentially*, iteratively constructing our attribute subset. Suppose at step k, the set of attributes selected so far is $\mathcal{A}_k^S$, $\mathcal{A}_k^{-S} = \mathcal{A} \setminus \mathcal{A}_k^S$ is the set difference between the original set of inputs $\mathcal{A}$ and $\mathcal{A}_k^S$. At step $k+1$ of forward selection search, these methods select the attribute X_{k+1} which maximizes the given score in Equation (10), Equation (11) or Equation (12). Then the attribute sets can be updated as $\mathcal{A}_{k+1}^S \leftarrow \mathcal{A}_k^S \cup \{X_{k+1}\}$, $\mathcal{A}_{k+1}^{-S} \leftarrow \mathcal{A}_k^{-S} \setminus \{X_{k+1}\}$. Initially, the set $\mathcal{A}^S$ is empty. So they select the attribute $\arg\max_{X \in \mathcal{A}} I(X;Y)$, which is with maximal mutual information with respect to the class variable. The procedure terminates when the set $\mathcal{A}_{-S}$ becomes empty.

3.4. Cross Validation Risk Minimization

Since the model space has been built, next we need select the best model in this space. A natural idea is to apply these models to the training examples and select the model with the best accuracy. However, this will cause the over fitting problem as the models have been trained and tested on the same examples. Low error rate on the training data set does not mean low error rate on the testing data set. The more practical way is to use only part of training examples to construct the models and leave the rest for testing and model selection. This is the idea of cross validation. In order to obtain the risks of different models in one pass learning through the training data, incremental leave-one-out cross validation [30] is adopted.

In the process of leave-one-out cross validation, the training set $\mathcal{D}$ is divided into a validation set (having only one instance) and an effective training set (having $|\mathcal{D}| - 1$ instances). Each instance in $\mathcal{D}$ can act as the validation instance in turn. At this time, the contribution of the instance to the models will be removed and the instance acts only as the validation instance. This realizes cross validation on the training set. During learning, no use of an instance in test set is made.

As the Root Mean Squared Error (RMSE) is a finer grained measure of the calibration of the probability estimates compared to zero-one loss, RMSE is adopted to measure the cross validation risk. Since the model that minimizes the empirical risk is searched for, we call the score Cross Validation Risk (CVR):

$$CVR = \sqrt{\frac{1}{|\mathcal{D}|}\sum_{i=1}^{|\mathcal{D}|}(1.0 - P_{\mathrm{TAN_s}}(y = y_i|\mathbf{x}_i))^2}\ , \tag{13}$$

Based on the above methodologies, we develop the training algorithm of attribute selective TAN, described in Algorithm 1. It involves two passes learning through the training set $\mathcal{D}$. One pass is to collect the information that is needed to form the table of joint frequencies of all combinations of 2 attributes values and the class label. The second pass is to evaluate all the models by leave-one-out cross validation.

Algorithm 1 Training algorithm of attribute selective TAN.

```
Form the table of joint frequencies of all combinations of 2 attribute values and the class label ▷ first pass through training data
Rank the attributes by Equations (8)–(12)
for instance inst ∈ D do                                   ▷ second pass through training data
    Remove inst from the frequency table
    Predict inst by all models in Equation (5)
    Accumulate the squared error for each model
    Add inst back to the frequency table
end for
Compute the CVR score for each model as in Equation (13)
Select the model with the lowest CVR
```

It could be found that this strategy can search through the model space in one more pass learning through the training data, thus it is efficient. Furthermore, local optima can be avoided in this strategy. This is different from those attribute selections that rely on hill climbing search [16], where multiple passes learning through training data might be involved and we can only get the local optima. In our strategy, if the search space could be expanded, the better model will be obtained.

From the training process in Algorithm 1, we could see that the space complexity of the table of joint frequencies of all combinations of 2 attributes values and the class label is $\mathcal{O}(c(av)^2)$, where v is the average number of values per attribute and c is the number of classes. Attribute selection will not require more memory. The time complexity consists of three parts. One is derivation of the frequencies required to populate the table, the time complexity of which is $\mathcal{O}(ta^2)$. The second part is the attribute ranking, the time

complexity of which is $\mathcal{O}(a^2)$. This part can be ignored given the first part. The last part is attribute selection in a second pass through the training data, the time complexity of which is $\mathcal{O}(tca)$, since for each example we need to compute the joint probability in Equation (4). So the overall time complexity is $\mathcal{O}(ta^2 + tca)$. The time complexity of classifying a single example is $\mathcal{O}(ca)$ in the worst-case scenario, because some attributes may be omitted after attribute selection.

4. Experiments and Analysis

4.1. Experimental Methodology

We have performed the experiments on 70 UCI data sets [31], covering a wide spectrum of number of instances (24–5,749,132), attributes (3–166) and classes (2–50), which allows us to examine the performance of the proposed algorithm on data sets with various characteristics. Table 1 lists the data sets, including the name of the data set, the number of instances, attributes, and classes. Note that the data sets have been listed in the ascending order of number of instances.

Table 1. Data sets.

No.	Name	Inst	Att	Class	No.	Name	Inst	Att	Class
1	contact-lenses	24	4	3	36	tic-tac-toe	958	9	2
2	lung-cancer	32	56	3	37	vowel	990	13	11
3	labor-negotiations	57	16	2	38	german	1000	20	2
4	post-operative	90	8	3	39	led	1000	7	10
5	zoo	101	16	7	40	contraceptive-mc	1473	9	3
6	promoters	106	57	2	41	yeast	1484	8	10
7	echocardiogram	131	6	2	42	volcanoes	1520	3	4
8	lymphography	148	18	4	43	car	1728	6	4
9	iris	150	4	3	44	segment	2310	19	7
10	teaching-ae	151	5	3	45	hypothyroid	3163	25	2
11	hepatitis	155	19	2	46	splice-c4.5	3177	60	3
12	wine	178	13	3	47	kr-vs-kp	3196	36	2
13	autos	205	25	7	48	abalone	4177	8	3
14	sonar	208	60	2	49	spambase	4601	57	2
15	glass-id	214	9	3	50	phoneme	5438	7	50
16	new-thyroid	215	5	3	51	wall-following	5456	24	4
17	audio	226	69	24	52	page-blocks	5473	10	5
18	hungarian	294	13	2	53	optdigits	5620	64	10
19	heart-disease-c	303	13	2	54	satellite	6435	36	6
20	haberman	306	3	2	55	musk2	6598	166	2
21	primary-tumor	339	17	22	56	mushrooms	8124	22	2
22	ionosphere	351	34	2	57	thyroid	9169	29	20
23	dermatology	366	34	6	58	pendigits	10,992	16	10
24	horse-colic	368	21	2	59	sign	12,546	8	3
25	house-votes-84	435	16	2	60	nursery	12,960	8	5
26	cylinder-bands	540	39	2	61	magic	19,020	10	2
27	chess	551	39	2	62	letter-recog	20,000	16	26
28	syncon	600	60	6	63	adult	48,842	14	2
29	balance-scale	625	4	3	64	shuttle	58,000	9	7
30	soybean	683	35	19	65	connect-4	67,557	42	3
31	credit-a	690	15	2	66	waveform	100,000	21	3
32	breast-cancer-w	699	9	2	67	localization	164,860	5	11
33	pima-ind-diabetes	768	8	2	68	census-income	299,285	41	2
34	vehicle	846	18	4	69	poker-hand	1,025,010	10	10
35	anneal	898	38	6	70	donation	5,749,132	11	2

The experiments has been done on a Linux HPC cluster which has 4 nodes each with 64 GB RAM. The experimental system is implemented in C++. In our experimental system, several different strategies from the Weka software [32] were adopted, namely:

1. Missing values are considered as a distinct value rather than replaced with modes and means for nominal and numeric attributes as in the Weka software.
2. Root mean squared error is calculated exclusively on the true class label. This is different from Weka's implementation, where all class labels are considered.

The base probabilities are estimated using m-estimation ($m = 1$) [33]. 5-bin equal frequency discretization is performed to discretize the numeric attributes as in [34]. All the algorithms have been run on the data sets in the 10-fold cross validation mode.

In the experiments, we will compare STAN with regular TAN. According to different attributes ranking strategies described in Section 3.3, we develop five versions of STAN, namely STAN$^{\mathrm{MI}}$, STAN$^{\mathrm{SU}}$, STAN$^{\mathrm{JMI}}$, STAN$^{\mathrm{CMIM}}$ and STAN$^{\mathrm{MRMR}}$. It is worthwhile to note that in the implementation of STAN$^{\mathrm{MRMR}}$, we use the following criterion instead of Equation (10) as suggested by the authors [24]:

$$X_{k+1}^{\mathrm{MRMR}} = \arg\max_{X \in \mathcal{A}_k^{-S}} \left\{ \frac{I(X;Y)}{\frac{1}{k}\sum_{X' \in \mathcal{A}_k^S} I(X;X') + 0.01} \right\}, \tag{14}$$

where 0.01 is added so as to avoid to be divided by zero. We also compare the best version of STAN with state-of-the-art one-dependence BNCs as AODE [5] and KDB$_1$ (KDB where $k = 1$).

4.2. Win/Draw/Loss Analysis

In this subsection, we demonstrate the classification performance of the proposed algorithms. Two commonly used performance measures are reported, namely Zero-one Loss (ZOL) and Root Mean Squared Error (RMSE). ZOL is the proportion of instances that are misclassified. RMSE is squared root of the mean squared probability that the testing example is misclassified, which is the difference between 1.0 and the probability estimated by the algorithm for the true class for the testing example.

Tables A1 and A2 in the Appendix A provide the detailed ZOL and RMSE results of eight algorithms on 70 data sets. In order to present a brief summary of the comparison of different algorithms, statistical win/draw/loss records in terms of the above performance measures are reported in Tables 2 and 3.

The win/draw/loss record indicates the frequency of one algorithm wins, draws with or loses to another algorithm with respect to the specified measure on 70 data sets. For example, win/draw/loss of STAN$^{\mathrm{MI}}$ against TAN with respect to ZOL is 26/22/22, which means STAN$^{\mathrm{MI}}$ obtains lower ZOL than TAN on 26 data sets, the same ZOL as TAN on 22 data sets and greater ZOL than TAN on 22 data sets.

To decide whether two comparing algorithms have the equal chances of win, a standard binomial sign test [35] is applied to these records. Given the null hypothesis that wins and losses are equiprobable, the binomial test indicates the probability of observing the specified numbers of win and loss. In our analysis, the number of draws is divided equally to the number of wins and losses. If the number of draws is an odd number, we ignore one. We reject the hypothesis and consider the difference between the two algorithms significant if the p value is less than the critical value 0.05, which is in bold font. The p value we reported is the outcome of a one-tailed test. For example, p value of STAN$^{\mathrm{MI}}$ against TAN is 0.3601, which means the probability of 37 $(26 + 22/2)$ wins in 70 comparisons is 0.3601 according to the binomial distribution. Since 0.3601 is greater than 0.5, we can draw a conclusion that the difference between STAN$^{\mathrm{MI}}$ and TAN is not significant, although STAN$^{\mathrm{MI}}$ obtains lower ZOL than TAN more often than the reverse.

We first compare different versions of STAN with TAN. From Table 2, we could find that relative to TAN, STAN$^{\mathrm{MI}}$ achieves lower error almost as often as higher. STAN$^{\mathrm{SU}}$,

STAN^{JMI} and STAN^{CMIM} deliver lower error more often than TAN, only significantly so with respect to STAN^{JMI} and STAN^{CMIM} on RMSE. STAN^{MRMR} reduces both zero-one loss and RMSE significantly often relative to TAN. We could conclude that along with the ranking strategy MRMR, STAN^{MRMR} achieves the best performance among the 5 five improvements.

We also present the scatter plot of STAN^{MRMR} to TAN in terms of ZOL in Figure 2. The points below the diagonal represent the data sets on which STAN^{MRMR} achieves lower ZOL than TAN. It could be found that STAN^{MRMR} provides consistently better predictions than regular TAN in a statistically way.

Table 2. Win/draw/loss records of different versions of STAN vs. TAN in terms of ZOL and RMSE on 70 data sets.

	STAN^{MI} vs. TAN		STAN^{SU} vs. TAN		STAN^{JMI} vs. TAN	
	win/draw/loss	p	win/draw/loss	p	win/draw/loss	p
ZOL	26/22/22	0.3601	29/20/21	0.2015	22/29/19	0.4050
RMSE	28/21/21	0.2352	32/19/19	0.0740	34/20/16	**0.0207**
	STAN^{CMIM} vs. TAN		STAN^{MRMR} vs. TAN			
	win/draw/loss	p	win/draw/loss	p		
ZOL	30/21/19	0.1142	34/22/14	**0.0112**		
RMSE	34/18/18	**0.0361**	38/17/15	**0.0038**		

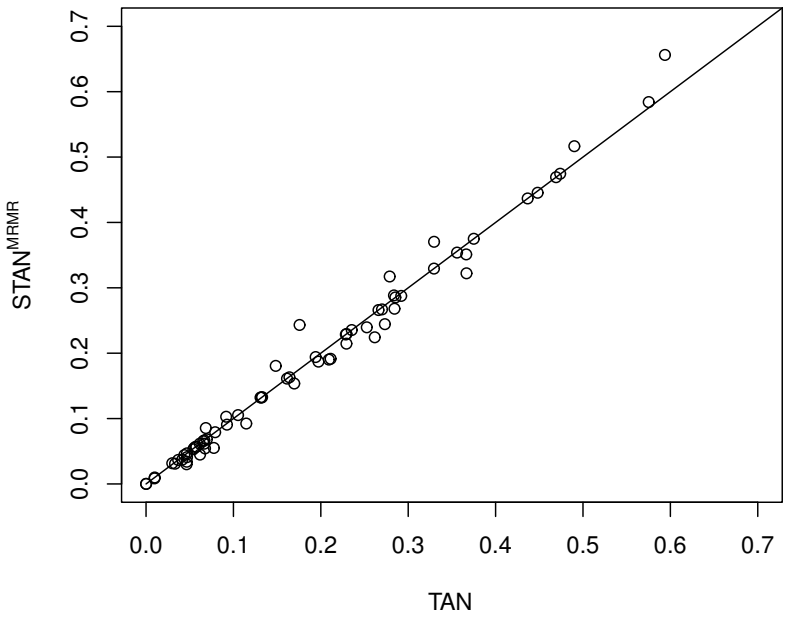


Figure 2. Scatter plot of STAN^{MRMR} to TAN in terms of ZOL

Next, we compare STAN^{MRMR} with state-of-the-art one-dependence Bayesian classifiers, AODE [5] and KDB [6]. We use the version of KDB where $k = 1$, simplified as KDB$_1$. The win/draw/loss results are summarized in Table 3. It could be found that STAN^{MRMR} achieves lower error almost as often as higher relative to AODE, while it obtains lower zero-one-loss and RMSE more often than KDB$_1$ than the reverse.

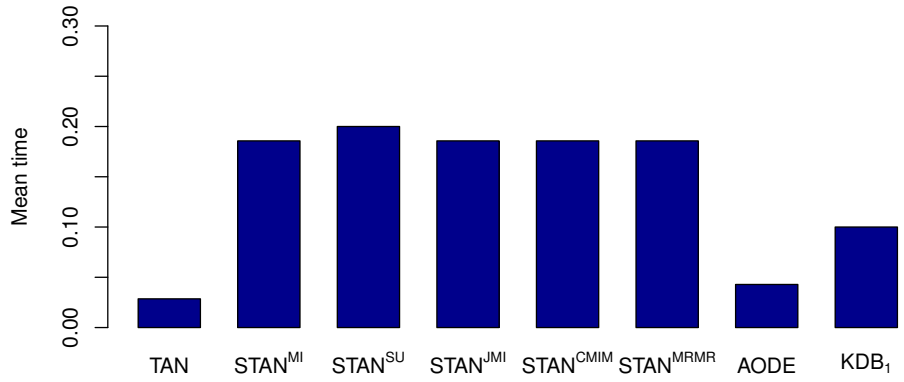
Table 3. Win/draw/loss records of $STAN^{MRMR}$ vs. AODE and KDB_1 in terms of ZOL and RMSE on 70 data sets.

	$STAN^{MRMR}$ vs. AODE		**$STAN^{MRMR}$ vs. KDB_1**	
	Win/Draw/Loss	***p***	**Win/Draw/Loss**	***p***
ZOL	33/1/36	0.4050	40/7/23	**0.0266**
RMSE	31/0/39	0.2015	45/1/24	**0.0077**

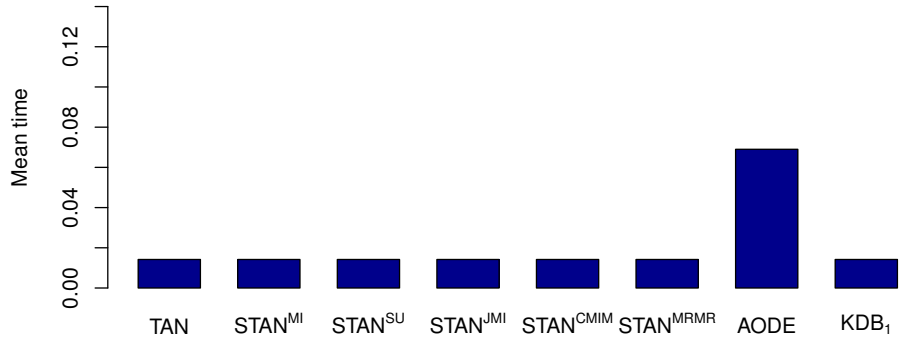
4.3. Analysis of Training and Classification Time

In this subsection, we will compare the averaged training and classification time of the proposed algorithms. The averaged training and classification time of all the 8 algorithms have been plotted in Figure 3.

It could be found that selective TAN requires more training time than regular TAN. This could be explained that selective TAN involves one more pass learning through the training data. While the difference between various versions of STAN is not significant. Five STAN algorithms require more training time than AODE and KDB_1.



(**a**) Training time



(**b**) Classification time

Figure 3. Averaged (**a**) training time and (**b**) classification time of all algorithms on 70 datasets (seconds).

As far as the classification time is concerned, five STAN algorithms achieve the same results as regular TAN. As the classification process uses less attributes in STAN than in regular TAN, the classification times are expected to be less than regular TAN. However, the plot does not indicate this trend. The deep observation of the classification times of different algorithms on each data set shows that the classification times on most data sets are 0 due

to the limited number of instances in those data sets. AODE require more classification time since it needs to classify the test instance by multiple one-dependence estimators.

5. Discussion

In this paper, we propose an attribute Selective Tree-Augmented Naive Bayes (STAN) algorithm, which builds a sequence of approximate models by adding one attribute at a time to the previous model and searches the model space to minimize the cross validation risk. The extensive experiments on 70 UCI data sets demonstrats that STAN achieves superior performance while maintaining the efficiency and simplicity. The conclusions are summarized as follows:

- STAN algorithms with different ranking strategies achieve superior classification performance than regular TAN at the cost of a modest increase in training time.
- MRMR ranking strategy achieves the best classification performance compared to other ranking strategies, and the advantage over regular TAN is significant.
- STAN with MRMR ranking strategy is comparable with AODE and superior to KDB_1 in terms of accuracy, while requires less classification time than AODE.

As the cross validation risk minimization provides an efficient search in the model space, expansion of the space would be expected to produce better models. So in the future it is worthwhile to expand the model space by varying the dependence level so as to find more practical model.

Author Contributions: Conceptualization, S.C.; methodology, S.C.; software, S.C.; validation, L.L.; formal analysis, S.C.; investigation, S.C.; resources, S.C.; data curation, S.C.; writing—original draft preparation, S.C.; writing—review and editing, S.C.; visualization, Z.Z.; supervision, S.C.; project administration, S.C.; funding acquisition, S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sets used in this manuscript can be found in UCI data repository (http://archive.ics.uci.edu/ml (accessed on 5 September 2021)).

Acknowledgments: This research has been supported in part by the NAU Educational Technology Center through the use of the NAU HPC Cluster.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. ZOL results of 8 algorithms on 70 data sets.

Dataset	TAN	STAN^{MI}	STAN^{SU}	STAN^{JMI}	STAN^{CMIM}	STAN^{MRMR}	AODE	KDB$_1$
contact-lenses	0.3750 ± 0.3758	0.3750 ± 0.3425	0.3750 ± 0.3425	0.3750 ± 0.3425	0.3750 ± 0.3425	0.3750 ± 0.3425	0.4167 ± 0.3574	0.2917 ± 0.3543
lung-cancer	0.5938 ± 0.2265	0.6875 ± 0.2641	0.7188 ± 0.2502	0.5625 ± 0.3289	0.6250 ± 0.2605	0.6562 ± 0.2733	0.4688 ± 0.2885	0.5938 ± 0.3082
labor-negotiations	0.1053 ± 0.1234	0.1053 ± 0.1234	0.0877 ± 0.1272	0.1053 ± 0.1234	0.1228 ± 0.2090	0.1053 ± 0.1234	0.0526 ± 0.0675	0.1053 ± 0.1146
post-operative	0.3667 ± 0.2075	0.3222 ± 0.2117	0.3444 ± 0.2124	0.3111 ± 0.1663	0.3111 ± 0.1663	0.3222 ± 0.2263	0.3444 ± 0.1882	0.3444 ± 0.1748
zoo	0.0099 ± 0.0527	0.0099 ± 0.0527	0.0099 ± 0.0527	0.0099 ± 0.0527	0.0198 ± 0.0542	0.0099 ± 0.0527	0.0198 ± 0.0384	0.0495 ± 0.0614
promoters	0.1321 ± 0.1036	0.1792 ± 0.1307	0.1887 ± 0.1251	0.1792 ± 0.1280	0.1604 ± 0.1072	0.1321 ± 0.1108	0.1038 ± 0.0648	0.1321 ± 0.0891
echocardiogram	0.3664 ± 0.1549	0.3893 ± 0.1191	0.3664 ± 0.0922	0.3969 ± 0.0998	0.3969 ± 0.0998	0.3511 ± 0.1132	0.3435 ± 0.1143	0.3664 ± 0.1511
lymphography	0.1757 ± 0.1003	0.1622 ± 0.1007	0.1689 ± 0.1005	0.1622 ± 0.1007	0.1689 ± 0.1047	0.2432 ± 0.1177	0.1486 ± 0.0991	0.1757 ± 0.0791
iris	0.0667 ± 0.0632	0.0667 ± 0.0632	0.0667 ± 0.0632	0.0667 ± 0.0632	0.0667 ± 0.0632	0.0667 ± 0.0632	0.0600 ± 0.0655	0.0733 ± 0.0505
teaching-ae	0.4901 ± 0.1245	0.4901 ± 0.1245	0.4901 ± 0.1245	0.5232 ± 0.1216	0.5232 ± 0.1216	0.5166 ± 0.1436	0.4834 ± 0.1179	0.4834 ± 0.1079
hepatitis	0.1484 ± 0.1280	0.1548 ± 0.1246	0.1548 ± 0.1264	0.1484 ± 0.1280	0.1484 ± 0.1264	0.1806 ± 0.1236	0.1935 ± 0.1244	0.2194 ± 0.1205
wine	0.0618 ± 0.0643	0.0562 ± 0.0534	0.0562 ± 0.0534	0.0618 ± 0.0647	0.0674 ± 0.0611	0.0449 ± 0.0532	0.0281 ± 0.0404	0.0674 ± 0.0633
autos	0.2293 ± 0.1374	0.1951 ± 0.1278	0.2000 ± 0.1118	0.2098 ± 0.1067	0.1951 ± 0.1162	0.2146 ± 0.1230	0.2537 ± 0.1104	0.2293 ± 0.1374
sonar	0.2788 ± 0.0840	0.3029 ± 0.1086	0.3029 ± 0.1086	0.2981 ± 0.0925	0.2692 ± 0.1032	0.3173 ± 0.1301	0.1394 ± 0.0888	0.2548 ± 0.0914
glass-id	0.2617 ± 0.0944	0.2664 ± 0.0923	0.2290 ± 0.0757	0.2523 ± 0.0944	0.2523 ± 0.0944	0.2243 ± 0.0671	0.1589 ± 0.0576	0.2383 ± 0.0720
new-thyroid	0.0791 ± 0.0647	0.0791 ± 0.0647	0.0791 ± 0.0647	0.0791 ± 0.0647	0.0791 ± 0.0647	0.0791 ± 0.0647	0.0512 ± 0.0544	0.0651 ± 0.0454
audio	0.2920 ± 0.0926	0.3053 ± 0.0676	0.3053 ± 0.0676	0.3009 ± 0.0851	0.3009 ± 0.0934	0.2876 ± 0.0821	0.2301 ± 0.0649	0.3097 ± 0.1054
hungarian	0.1973 ± 0.0606	0.2041 ± 0.0613	0.2109 ± 0.0708	0.1973 ± 0.0524	0.1905 ± 0.0511	0.1871 ± 0.0789	0.1429 ± 0.0676	0.2075 ± 0.0625
heart-disease-c	0.2112 ± 0.1005	0.2211 ± 0.1154	0.2244 ± 0.1037	0.2178 ± 0.1020	0.2079 ± 0.1081	0.1914 ± 0.0820	0.1848 ± 0.1067	0.2178 ± 0.1428
haberman	0.2843 ± 0.1023	0.2778 ± 0.0868	0.2778 ± 0.0868	0.2745 ± 0.0851	0.2745 ± 0.0851	0.2680 ± 0.0985	0.2712 ± 0.1188	0.2778 ± 0.1024
primary-tumor	0.5752 ± 0.0960	0.5841 ± 0.1188	0.5841 ± 0.1188	0.5841 ± 0.1188	0.5782 ± 0.1209	0.5841 ± 0.1184	0.5162 ± 0.0984	0.5841 ± 0.1119
ionosphere	0.0684 ± 0.0510	0.0741 ± 0.0453	0.0769 ± 0.0448	0.0741 ± 0.0542	0.0712 ± 0.0409	0.0855 ± 0.0428	0.0826 ± 0.0405	0.0684 ± 0.0441
dermatology	0.0464 ± 0.0390	0.0383 ± 0.0345	0.0437 ± 0.0359	0.0410 ± 0.0287	0.0437 ± 0.0378	0.0301 ± 0.0282	0.0219 ± 0.0275	0.0301 ± 0.0258
horse-colic	0.2092 ± 0.0629	0.1875 ± 0.0524	0.1793 ± 0.0567	0.1793 ± 0.0715	0.1685 ± 0.0618	0.1902 ± 0.0604	0.2038 ± 0.0590	0.2120 ± 0.0615
house-votes-84	0.0552 ± 0.0375	0.0644 ± 0.0386	0.0644 ± 0.0386	0.0552 ± 0.0315	0.0529 ± 0.0404	0.0552 ± 0.0368	0.0529 ± 0.0346	0.0690 ± 0.0353
cylinder-bands	0.3296 ± 0.0719	0.3833 ± 0.0730	0.3796 ± 0.0821	0.3722 ± 0.0634	0.3759 ± 0.0677	0.3704 ± 0.0774	0.1611 ± 0.0421	0.2074 ± 0.0575
chess	0.0926 ± 0.0492	0.0907 ± 0.0509	0.0907 ± 0.0509	0.0944 ± 0.0553	0.0907 ± 0.0509	0.0907 ± 0.0509	0.1053 ± 0.0631	0.0998 ± 0.0354
syncon	0.0300 ± 0.0249	0.0283 ± 0.0241	0.0283 ± 0.0241	0.0317 ± 0.0266	0.0267 ± 0.0235	0.0317 ± 0.0291	0.0200 ± 0.0163	0.0200 ± 0.0156
balance-scale	0.1328 ± 0.0156	0.1328 ± 0.0156	0.1328 ± 0.0156	0.1328 ± 0.0156	0.1328 ± 0.0156	0.1328 ± 0.0156	0.1120 ± 0.0159	0.1424 ± 0.0307
soybean	0.0469 ± 0.0136	0.0586 ± 0.0195	0.0571 ± 0.0180	0.0469 ± 0.0158	0.0454 ± 0.0100	0.0410 ± 0.0095	0.0542 ± 0.0184	0.0644 ± 0.0205
credit-a	0.1696 ± 0.0370	0.1667 ± 0.0394	0.1623 ± 0.0374	0.1739 ± 0.0460	0.1696 ± 0.0444	0.1536 ± 0.0377	0.1261 ± 0.0210	0.1696 ± 0.0417
breast-cancer-w	0.0415 ± 0.0273	0.0443 ± 0.0252	0.0429 ± 0.0271	0.0415 ± 0.0271	0.0386 ± 0.0207	0.0372 ± 0.0237	0.0386 ± 0.0248	0.0486 ± 0.0181

Table A1. *Cont.*

Dataset	TAN	STAN^{MI}	STAN^{SU}	STAN^{JMI}	STAN^{CMIM}	STAN^{MRMR}	AODE	KDB$_1$
pima-ind-diabetes	0.2526 ± 0.0509	0.2487 ± 0.0416	0.2487 ± 0.0416	0.2409 ± 0.0505	0.2461 ± 0.0480	0.2396 ± 0.0550	0.2513 ± 0.0636	0.2578 ± 0.0583
vehicle	0.2837 ± 0.0603	0.3014 ± 0.0505	0.3014 ± 0.0505	0.3121 ± 0.0479	0.2837 ± 0.0570	0.2884 ± 0.0654	0.3132 ± 0.0563	0.3026 ± 0.0627
anneal	0.0468 ± 0.0182	0.0468 ± 0.0182	0.0468 ± 0.0182	0.0468 ± 0.0182	0.0468 ± 0.0182	0.0468 ± 0.0182	0.0735 ± 0.0232	0.0445 ± 0.0156
tic-tac-toe	0.2286 ± 0.0395	0.2286 ± 0.0395	0.2286 ± 0.0395	0.2286 ± 0.0395	0.2286 ± 0.0395	0.2286 ± 0.0395	0.2683 ± 0.0432	0.2463 ± 0.0382
vowel	0.0667 ± 0.0259	0.0616 ± 0.0284	0.0616 ± 0.0284	0.0646 ± 0.0270	0.0707 ± 0.0354	0.0616 ± 0.0284	0.0808 ± 0.0296	0.2162 ± 0.0272
german	0.2700 ± 0.0515	0.2750 ± 0.0411	0.2760 ± 0.0470	0.2770 ± 0.0653	0.2740 ± 0.0604	0.2670 ± 0.0398	0.2410 ± 0.0535	0.2660 ± 0.0634
led	0.2660 ± 0.0569	0.2660 ± 0.0569	0.2660 ± 0.0569	0.2660 ± 0.0569	0.2660 ± 0.0569	0.2660 ± 0.0569	0.2700 ± 0.0604	0.2640 ± 0.0603
contraceptive-mc	0.4739 ± 0.0345	0.4800 ± 0.0328	0.4779 ± 0.0318	0.4793 ± 0.0333	0.4739 ± 0.0234	0.4745 ± 0.0266	0.4671 ± 0.0455	0.4684 ± 0.0276
yeast	0.4481 ± 0.0360	0.4481 ± 0.0360	0.4461 ± 0.0324	0.4481 ± 0.0360	0.4481 ± 0.0360	0.4454 ± 0.0322	0.4205 ± 0.0402	0.4394 ± 0.0326
volcanoes	0.3559 ± 0.0250	0.3539 ± 0.0276	0.3539 ± 0.0276	0.3533 ± 0.0294	0.3533 ± 0.0294	0.3539 ± 0.0276	0.3539 ± 0.0331	0.3520 ± 0.0258
car	0.0567 ± 0.0182	0.0567 ± 0.0182	0.0567 ± 0.0182	0.0567 ± 0.0182	0.0567 ± 0.0182	0.0567 ± 0.0182	0.0845 ± 0.0193	0.0567 ± 0.0182
segment	0.0615 ± 0.0142	0.0610 ± 0.0133	0.0610 ± 0.0133	0.0610 ± 0.0133	0.0610 ± 0.0133	0.0615 ± 0.0130	0.0563 ± 0.0091	0.0567 ± 0.0158
hypothyroid	0.0332 ± 0.0126	0.0319 ± 0.0110	0.0322 ± 0.0098	0.0326 ± 0.0104	0.0316 ± 0.0106	0.0310 ± 0.0095	0.0348 ± 0.0118	0.0338 ± 0.0137
splice-c4.5	0.0466 ± 0.0129	0.0349 ± 0.0089	0.0349 ± 0.0089	0.0349 ± 0.0102	0.0318 ± 0.0078	0.0340 ± 0.0088	0.0375 ± 0.0087	0.0482 ± 0.0152
kr-vs-kp	0.0776 ± 0.0228	0.0569 ± 0.0186	0.0579 ± 0.0187	0.0569 ± 0.0186	0.0607 ± 0.0145	0.0551 ± 0.0142	0.0854 ± 0.0187	0.0544 ± 0.0171
abalone	0.4692 ± 0.0285	0.4692 ± 0.0285	0.4692 ± 0.0285	0.4692 ± 0.0285	0.4690 ± 0.0279	0.4692 ± 0.0285	0.4551 ± 0.0214	0.4656 ± 0.0237
spambase	0.0696 ± 0.0106	0.0689 ± 0.0115	0.0685 ± 0.0118	0.0696 ± 0.0106	0.0689 ± 0.0112	0.0682 ± 0.0114	0.0635 ± 0.0114	0.0702 ± 0.0121
phoneme	0.2733 ± 0.0177	0.2413 ± 0.0119	0.2413 ± 0.0119	0.2413 ± 0.0119	0.2413 ± 0.0119	0.2444 ± 0.0119	0.2100 ± 0.0144	0.2120 ± 0.0123
wall-following	0.1147 ± 0.0116	0.0872 ± 0.0092	0.0872 ± 0.0092	0.0867 ± 0.0097	0.0861 ± 0.0108	0.0924 ± 0.0110	0.1514 ± 0.0101	0.1043 ± 0.0094
page-blocks	0.0541 ± 0.0100	0.0530 ± 0.0081	0.0530 ± 0.0081	0.0541 ± 0.0100	0.0550 ± 0.0099	0.0530 ± 0.0081	0.0502 ± 0.0066	0.0590 ± 0.0102
optdigits	0.0438 ± 0.0064	0.0441 ± 0.0064	0.0441 ± 0.0064	0.0441 ± 0.0064	0.0443 ± 0.0068	0.0441 ± 0.0067	0.0283 ± 0.0095	0.0454 ± 0.0070
satellite	0.1310 ± 0.0126	0.1321 ± 0.0118	0.1321 ± 0.0118	0.1318 ± 0.0126	0.1340 ± 0.0135	0.1322 ± 0.0132	0.1301 ± 0.0131	0.1392 ± 0.0135
musk2	0.0917 ± 0.0086	0.1003 ± 0.0142	0.1073 ± 0.0165	0.0996 ± 0.0146	0.0997 ± 0.0187	0.1028 ± 0.0127	0.1511 ± 0.0101	0.0867 ± 0.0097
mushrooms	0.0001 ± 0.0004	0.0001 ± 0.0004	0.0001 ± 0.0004	0.0001 ± 0.0004	0.0000 ± 0.0000	0.0001 ± 0.0004	0.0002 ± 0.0005	0.0006 ± 0.0009
thyroid	0.2294 ± 0.0111	0.2294 ± 0.0111	0.2301 ± 0.0121	0.2294 ± 0.0111	0.2294 ± 0.0111	0.2294 ± 0.0111	0.2421 ± 0.0136	0.2319 ± 0.0146
pendigits	0.0576 ± 0.0064	0.0576 ± 0.0064	0.0576 ± 0.0064	0.0552 ± 0.0056	0.0544 ± 0.0058	0.0568 ± 0.0064	0.0254 ± 0.0029	0.0529 ± 0.0066
sign	0.2853 ± 0.0094	0.2853 ± 0.0094	0.2853 ± 0.0094	0.2853 ± 0.0094	0.2853 ± 0.0094	0.2853 ± 0.0094	0.2960 ± 0.0119	0.3055 ± 0.0140
nursery	0.0654 ± 0.0062	0.0654 ± 0.0062	0.0654 ± 0.0062	0.0654 ± 0.0062	0.0654 ± 0.0062	0.0654 ± 0.0062	0.0733 ± 0.0059	0.0654 ± 0.0061
magic	0.1613 ± 0.0076	0.1611 ± 0.0086	0.1611 ± 0.0086	0.1613 ± 0.0076	0.1613 ± 0.0076	0.1611 ± 0.0076	0.1726 ± 0.0084	0.1759 ± 0.0107
letter-recog	0.1941 ± 0.0085	0.1941 ± 0.0085	0.1941 ± 0.0085	0.1941 ± 0.0085	0.1941 ± 0.0085	0.1941 ± 0.0085	0.1514 ± 0.0089	0.1920 ± 0.0112
adult	0.1641 ± 0.0037	0.1609 ± 0.0040	0.1609 ± 0.0040	0.1635 ± 0.0034	0.1642 ± 0.0037	0.1631 ± 0.0045	0.1679 ± 0.0032	0.1638 ± 0.0044
shuttle	0.0097 ± 0.0013	0.0085 ± 0.0013	0.0085 ± 0.0013	0.0085 ± 0.0013	0.0085 ± 0.0013	0.0085 ± 0.0013	0.0101 ± 0.0010	0.0163 ± 0.0012
connect-4	0.2354 ± 0.0050	0.2354 ± 0.0050	0.2354 ± 0.0050	0.2354 ± 0.0050	0.2354 ± 0.0050	0.2354 ± 0.0050	0.2422 ± 0.0047	0.2406 ± 0.0030

Table A1. *Cont.*

Dataset	TAN	STAN^{MI}	STAN^{SU}	STAN^{JMI}	STAN^{CMIM}	STAN^{MRMR}	AODE	KDB$_1$
waveform	0.0368 ± 0.0015	0.0370 ± 0.0014	0.0370 ± 0.0014	0.0369 ± 0.0014	0.0369 ± 0.0014	0.0367 ± 0.0015	0.0343 ± 0.0008	0.0396 ± 0.0021
localization	0.4367 ± 0.0033	0.4367 ± 0.0033	0.4367 ± 0.0033	0.4367 ± 0.0033	0.4367 ± 0.0033	0.4367 ± 0.0033	0.4333 ± 0.0027	0.4642 ± 0.0040
census-income	0.0675 ± 0.0016	0.0585 ± 0.0020	0.0571 ± 0.0010	0.0567 ± 0.0010	0.0599 ± 0.0011	0.0544 ± 0.0015	0.1106 ± 0.0015	0.0667 ± 0.0014
poker-hand	0.3295 ± 0.0015	0.3294 ± 0.0015	0.3294 ± 0.0015	0.3294 ± 0.0015	0.3294 ± 0.0015	0.3294 ± 0.0015	0.4812 ± 0.0028	0.3291 ± 0.0012
donation	0.0001 ± 0.0000	0.0001 ± 0.0000	0.0001 ± 0.0000	0.0001 ± 0.0000	0.0001 ± 0.0000	0.0001 ± 0.0000	0.0002 ± 0.0000	0.0001 ± 0.0000

Table A2. RMSE results of 8 algorithms on 70 data sets.

Dataset	TAN	STAN^{MI}	STAN^{SU}	STAN^{JMI}	STAN^{CMIM}	STAN^{MRMR}	AODE	KDB$_1$
contact-lenses	0.6077 ± 0.1831	0.5438 ± 0.2091	0.5438 ± 0.2091	0.5635 ± 0.2278	0.5438 ± 0.2091	0.5635 ± 0.2278	0.5226 ± 0.2221	0.5024 ± 0.2104
lung-cancer	0.7623 ± 0.1357	0.8044 ± 0.1515	0.7955 ± 0.1412	0.6807 ± 0.2643	0.7364 ± 0.1709	0.7690 ± 0.1463	0.6614 ± 0.2444	0.7523 ± 0.2928
labor-negotiations	0.2935 ± 0.1975	0.2988 ± 0.2081	0.2847 ± 0.2132	0.2877 ± 0.1972	0.3131 ± 0.2514	0.2915 ± 0.2033	0.2104 ± 0.1455	0.3014 ± 0.1907
post-operative	0.5340 ± 0.1393	0.5153 ± 0.1354	0.5206 ± 0.1241	0.5017 ± 0.1150	0.5017 ± 0.1150	0.5133 ± 0.1405	0.5136 ± 0.1059	0.5289 ± 0.1031
zoo	0.1309 ± 0.1131	0.1168 ± 0.1054	0.1168 ± 0.1054	0.1144 ± 0.1052	0.1477 ± 0.1144	0.1397 ± 0.1139	0.1344 ± 0.0935	0.1984 ± 0.1255
promoters	0.3264 ± 0.1659	0.3883 ± 0.1721	0.3895 ± 0.1698	0.3864 ± 0.1749	0.3702 ± 0.1647	0.3485 ± 0.1748	0.2795 ± 0.0940	0.3292 ± 0.1603
echocardiogram	0.5276 ± 0.1017	0.5144 ± 0.0890	0.4999 ± 0.0640	0.5073 ± 0.0741	0.5073 ± 0.0741	0.4986 ± 0.0693	0.4829 ± 0.0808	0.5288 ± 0.1034
lymphography	0.3813 ± 0.1227	0.3816 ± 0.1231	0.3891 ± 0.1250	0.3814 ± 0.1230	0.3874 ± 0.1175	0.4369 ± 0.0996	0.3274 ± 0.1395	0.3726 ± 0.1169
iris	0.2211 ± 0.1353	0.2211 ± 0.1353	0.2211 ± 0.1353	0.2211 ± 0.1353	0.2211 ± 0.1353	0.2211 ± 0.1353	0.2224 ± 0.1303	0.2273 ± 0.1252
teaching-ae	0.6189 ± 0.0671	0.6189 ± 0.0671	0.6189 ± 0.0671	0.6272 ± 0.0834	0.6272 ± 0.0834	0.6404 ± 0.0749	0.6105 ± 0.0684	0.6224 ± 0.0683
hepatitis	0.3434 ± 0.1479	0.3530 ± 0.1396	0.3409 ± 0.1326	0.3416 ± 0.1422	0.3475 ± 0.1459	0.3751 ± 0.1442	0.3711 ± 0.1079	0.4188 ± 0.1082
wine	0.2026 ± 0.1223	0.2020 ± 0.1179	0.2063 ± 0.1234	0.2142 ± 0.1218	0.2202 ± 0.1180	0.1923 ± 0.1029	0.1528 ± 0.1007	0.2210 ± 0.0927
autos	0.4725 ± 0.1291	0.4214 ± 0.1637	0.4241 ± 0.1350	0.4339 ± 0.1327	0.4312 ± 0.1356	0.4350 ± 0.1427	0.4760 ± 0.1102	0.4736 ± 0.1286
sonar	0.4856 ± 0.0890	0.5085 ± 0.1087	0.5085 ± 0.1087	0.5027 ± 0.0989	0.4805 ± 0.0920	0.5161 ± 0.1172	0.3349 ± 0.1109	0.4629 ± 0.0783
glass-id	0.4360 ± 0.0585	0.4504 ± 0.0608	0.4286 ± 0.0604	0.4381 ± 0.0594	0.4371 ± 0.0583	0.4170 ± 0.0587	0.3654 ± 0.0546	0.4199 ± 0.0650
new-thyroid	0.2554 ± 0.0991	0.2554 ± 0.0991	0.2554 ± 0.0991	0.2554 ± 0.0991	0.2554 ± 0.0991	0.2554 ± 0.0991	0.2221 ± 0.0850	0.2262 ± 0.0710
audio	0.5212 ± 0.0855	0.5168 ± 0.0663	0.5175 ± 0.0660	0.5151 ± 0.0803	0.5139 ± 0.0809	0.5136 ± 0.0672	0.4639 ± 0.0606	0.5294 ± 0.0939
hungarian	0.3895 ± 0.0711	0.3882 ± 0.0740	0.3816 ± 0.0684	0.3855 ± 0.0610	0.3870 ± 0.0548	0.3778 ± 0.0628	0.3506 ± 0.0845	0.3917 ± 0.0684
heart-disease-c	0.4177 ± 0.0861	0.4203 ± 0.0881	0.4159 ± 0.0778	0.4171 ± 0.0820	0.4152 ± 0.0810	0.3874 ± 0.0692	0.3605 ± 0.0844	0.4135 ± 0.0989
haberman	0.4433 ± 0.0759	0.4299 ± 0.0756	0.4299 ± 0.0756	0.4280 ± 0.0743	0.4280 ± 0.0743	0.4283 ± 0.0728	0.4402 ± 0.0820	0.4416 ± 0.0776
primary-tumor	0.7280 ± 0.0579	0.7272 ± 0.0574	0.7264 ± 0.0610	0.7272 ± 0.0574	0.7266 ± 0.0580	0.7258 ± 0.0572	0.6972 ± 0.0585	0.7250 ± 0.0589
ionosphere	0.2573 ± 0.1077	0.2616 ± 0.0991	0.2654 ± 0.0981	0.2596 ± 0.1044	0.2452 ± 0.0947	0.2638 ± 0.0941	0.2841 ± 0.0724	0.2434 ± 0.1047
dermatology	0.1826 ± 0.0695	0.1792 ± 0.0745	0.1786 ± 0.0719	0.1786 ± 0.0633	0.1878 ± 0.0793	0.1576 ± 0.0593	0.1145 ± 0.0617	0.1521 ± 0.0784

Table A2. *Cont.*

Dataset	TAN	STAN^{MI}	STAN^{SU}	STAN^{JMI}	STAN^{CMIM}	STAN^{MRMR}	AODE	KDB$_1$
horse-colic	0.4289 ± 0.0672	0.3829 ± 0.0585	0.3714 ± 0.0569	0.3746 ± 0.0593	0.3764 ± 0.0587	0.3879 ± 0.0520	0.4029 ± 0.0709	0.4185 ± 0.0597
house-votes-84	0.2181 ± 0.0792	0.2337 ± 0.0803	0.2337 ± 0.0803	0.2161 ± 0.0663	0.2123 ± 0.0789	0.2115 ± 0.0686	0.2016 ± 0.0736	0.2235 ± 0.0721
cylinder-bands	0.4405 ± 0.0420	0.4407 ± 0.0282	0.4393 ± 0.0280	0.4365 ± 0.0278	0.4423 ± 0.0281	0.4436 ± 0.0246	0.3656 ± 0.0451	0.4312 ± 0.0590
chess	0.2594 ± 0.0470	0.2589 ± 0.0477	0.2590 ± 0.0475	0.2613 ± 0.0495	0.2597 ± 0.0479	0.2590 ± 0.0475	0.2855 ± 0.0485	0.2671 ± 0.0384
syncon	0.1602 ± 0.0688	0.1608 ± 0.0807	0.1608 ± 0.0807	0.1651 ± 0.0748	0.1557 ± 0.0862	0.1617 ± 0.0800	0.1287 ± 0.0448	0.1271 ± 0.0686
balance-scale	0.3971 ± 0.0186	0.3971 ± 0.0186	0.3971 ± 0.0186	0.3971 ± 0.0186	0.3971 ± 0.0186	0.3971 ± 0.0186	0.3999 ± 0.0234	0.4014 ± 0.0200
soybean	0.2014 ± 0.0341	0.2139 ± 0.0365	0.2062 ± 0.0347	0.1914 ± 0.0294	0.1945 ± 0.0337	0.1828 ± 0.0265	0.2224 ± 0.0402	0.2206 ± 0.0436
credit-a	0.3704 ± 0.0443	0.3404 ± 0.0242	0.3377 ± 0.0396	0.3386 ± 0.0314	0.3371 ± 0.0334	0.3473 ± 0.0307	0.3164 ± 0.0387	0.3692 ± 0.0419
breast-cancer-w	0.1928 ± 0.0618	0.1877 ± 0.0544	0.1931 ± 0.0595	0.1794 ± 0.0566	0.1830 ± 0.0475	0.1746 ± 0.0527	0.1778 ± 0.0879	0.1951 ± 0.0461
pima-ind-diabetes	0.4225 ± 0.0442	0.4142 ± 0.0496	0.4142 ± 0.0496	0.4116 ± 0.0516	0.4114 ± 0.0501	0.4079 ± 0.0495	0.4071 ± 0.0438	0.4212 ± 0.0494
vehicle	0.4638 ± 0.0458	0.4691 ± 0.0389	0.4691 ± 0.0389	0.4706 ± 0.0387	0.4634 ± 0.0448	0.4611 ± 0.0425	0.4653 ± 0.0343	0.4637 ± 0.0433
anneal	0.1813 ± 0.0366	0.1813 ± 0.0366	0.1813 ± 0.0366	0.1813 ± 0.0366	0.1813 ± 0.0366	0.1813 ± 0.0366	0.2311 ± 0.0373	0.1815 ± 0.0330
tic-tac-toe	0.4023 ± 0.0269	0.4023 ± 0.0269	0.4023 ± 0.0269	0.4023 ± 0.0269	0.4023 ± 0.0269	0.4023 ± 0.0269	0.3995 ± 0.0212	0.4050 ± 0.0252
vowel	0.2316 ± 0.0407	0.2232 ± 0.0390	0.2232 ± 0.0390	0.2305 ± 0.0405	0.2366 ± 0.0481	0.2232 ± 0.0390	0.2593 ± 0.0347	0.4182 ± 0.0185
german	0.4389 ± 0.0476	0.4380 ± 0.0389	0.4374 ± 0.0416	0.4348 ± 0.0429	0.4385 ± 0.0354	0.4368 ± 0.0339	0.4147 ± 0.0305	0.4333 ± 0.0392
led	0.5000 ± 0.0376	0.5000 ± 0.0376	0.5000 ± 0.0376	0.5000 ± 0.0376	0.5000 ± 0.0376	0.5000 ± 0.0376	0.4970 ± 0.0364	0.4991 ± 0.0375
contraceptive-mc	0.5955 ± 0.0148	0.5970 ± 0.0131	0.5957 ± 0.0112	0.5969 ± 0.0132	0.5955 ± 0.0122	0.5965 ± 0.0123	0.5938 ± 0.0183	0.5923 ± 0.0164
yeast	0.6204 ± 0.0226	0.6204 ± 0.0226	0.6205 ± 0.0195	0.6204 ± 0.0226	0.6204 ± 0.0226	0.6201 ± 0.0188	0.6063 ± 0.0195	0.6144 ± 0.0201
volcanoes	0.5313 ± 0.0155	0.5324 ± 0.0146	0.5324 ± 0.0146	0.5322 ± 0.0144	0.5322 ± 0.0144	0.5324 ± 0.0146	0.5284 ± 0.0184	0.5297 ± 0.0168
car	0.2405 ± 0.0171	0.2405 ± 0.0171	0.2405 ± 0.0171	0.2405 ± 0.0171	0.2405 ± 0.0171	0.2405 ± 0.0171	0.3065 ± 0.0151	0.2404 ± 0.0170
segment	0.2215 ± 0.0255	0.2216 ± 0.0254	0.2216 ± 0.0254	0.2216 ± 0.0254	0.2216 ± 0.0254	0.2218 ± 0.0254	0.2069 ± 0.0143	0.2166 ± 0.0256
hypothyroid	0.1528 ± 0.0262	0.1448 ± 0.0195	0.1442 ± 0.0187	0.1467 ± 0.0187	0.1445 ± 0.0195	0.1447 ± 0.0192	0.1636 ± 0.0277	0.1517 ± 0.0268
splice-c4.5	0.1917 ± 0.0248	0.1670 ± 0.0150	0.1670 ± 0.0150	0.1650 ± 0.0153	0.1638 ± 0.0160	0.1640 ± 0.0141	0.1720 ± 0.0211	0.1944 ± 0.0245
kr-vs-kp	0.2358 ± 0.0223	0.2249 ± 0.0218	0.2230 ± 0.0214	0.2244 ± 0.0229	0.2206 ± 0.0215	0.2220 ± 0.0200	0.2658 ± 0.0155	0.2159 ± 0.0229
abalone	0.5635 ± 0.0080	0.5635 ± 0.0080	0.5635 ± 0.0080	0.5635 ± 0.0080	0.5634 ± 0.0079	0.5635 ± 0.0080	0.5576 ± 0.0077	0.5638 ± 0.0081
spambase	0.2377 ± 0.0187	0.2370 ± 0.0194	0.2366 ± 0.0196	0.2374 ± 0.0196	0.2367 ± 0.0197	0.2365 ± 0.0196	0.2282 ± 0.0212	0.2383 ± 0.0206
phoneme	0.5048 ± 0.0133	0.4789 ± 0.0104	0.4789 ± 0.0104	0.4789 ± 0.0104	0.4789 ± 0.0104	0.4799 ± 0.0101	0.4397 ± 0.0123	0.4385 ± 0.0127
wall-following	0.3113 ± 0.0146	0.2598 ± 0.0121	0.2598 ± 0.0121	0.2602 ± 0.0142	0.2658 ± 0.0149	0.2661 ± 0.0121	0.3677 ± 0.0136	0.2968 ± 0.0145
page-blocks	0.2127 ± 0.0211	0.2095 ± 0.0198	0.2095 ± 0.0198	0.2127 ± 0.0211	0.2141 ± 0.0210	0.2095 ± 0.0198	0.2024 ± 0.0110	0.2168 ± 0.0173
optdigits	0.1919 ± 0.0125	0.1924 ± 0.0127	0.1924 ± 0.0127	0.1924 ± 0.0127	0.1931 ± 0.0133	0.1922 ± 0.0129	0.1542 ± 0.0236	0.1968 ± 0.0162
satellite	0.3396 ± 0.0173	0.3403 ± 0.0165	0.3403 ± 0.0165	0.3407 ± 0.0172	0.3424 ± 0.0177	0.3406 ± 0.0176	0.3307 ± 0.0161	0.3479 ± 0.0195
musk2	0.2946 ± 0.0144	0.2961 ± 0.0110	0.2826 ± 0.0172	0.2982 ± 0.0115	0.2998 ± 0.0121	0.2762 ± 0.0159	0.3837 ± 0.0115	0.2847 ± 0.0153
mushrooms	0.0083 ± 0.0082	0.0083 ± 0.0082	0.0083 ± 0.0082	0.0083 ± 0.0082	0.0036 ± 0.0035	0.0083 ± 0.0082	0.0112 ± 0.0098	0.0188 ± 0.0155
thyroid	0.4156 ± 0.0103	0.4156 ± 0.0103	0.4158 ± 0.0106	0.4156 ± 0.0103	0.4156 ± 0.0103	0.4156 ± 0.0103	0.4334 ± 0.0109	0.4193 ± 0.0127

Table A2. *Cont.*

Dataset	TAN	STAN^{MI}	STAN^{SU}	STAN^{JMI}	STAN^{CMIM}	STAN^{MRMR}	AODE	KDB$_1$
pendigits	0.2140 ± 0.0130	0.2140 ± 0.0130	0.2140 ± 0.0130	0.2127 ± 0.0135	0.2116 ± 0.0133	0.2138 ± 0.0134	0.1420 ± 0.0047	0.2060 ± 0.0120
sign	0.4736 ± 0.0058	0.4736 ± 0.0058	0.4736 ± 0.0058	0.4736 ± 0.0058	0.4736 ± 0.0058	0.4736 ± 0.0058	0.4835 ± 0.0042	0.4911 ± 0.0073
nursery	0.2194 ± 0.0068	0.2194 ± 0.0068	0.2194 ± 0.0068	0.2194 ± 0.0068	0.2194 ± 0.0068	0.2194 ± 0.0068	0.2510 ± 0.0047	0.2193 ± 0.0068
magic	0.3437 ± 0.0068	0.3438 ± 0.0072	0.3438 ± 0.0072	0.3437 ± 0.0068	0.3437 ± 0.0068	0.3435 ± 0.0072	0.3505 ± 0.0079	0.3547 ± 0.0070
letter-recog	0.4120 ± 0.0085	0.4120 ± 0.0085	0.4120 ± 0.0085	0.4120 ± 0.0085	0.4120 ± 0.0085	0.4120 ± 0.0085	0.3755 ± 0.0092	0.4106 ± 0.0101
adult	0.3354 ± 0.0040	0.3322 ± 0.0037	0.3322 ± 0.0037	0.3339 ± 0.0035	0.3353 ± 0.0038	0.3335 ± 0.0040	0.3476 ± 0.0035	0.3345 ± 0.0037
shuttle	0.0907 ± 0.0046	0.0865 ± 0.0046	0.0865 ± 0.0046	0.0865 ± 0.0046	0.0865 ± 0.0046	0.0865 ± 0.0046	0.0944 ± 0.0033	0.1036 ± 0.0037
connect-4	0.4435 ± 0.0031	0.4435 ± 0.0031	0.4435 ± 0.0031	0.4435 ± 0.0031	0.4435 ± 0.0031	0.4435 ± 0.0031	0.4506 ± 0.0018	0.4480 ± 0.0022
waveform	0.1597 ± 0.0018	0.1597 ± 0.0018	0.1597 ± 0.0018	0.1595 ± 0.0021	0.1596 ± 0.0018	0.1593 ± 0.0018	0.1528 ± 0.0020	0.1684 ± 0.0051
localization	0.6321 ± 0.0014	0.6321 ± 0.0014	0.6321 ± 0.0014	0.6321 ± 0.0014	0.6321 ± 0.0014	0.6321 ± 0.0014	0.6520 ± 0.0010	0.6501 ± 0.0012
census-income	0.2247 ± 0.0025	0.2134 ± 0.0019	0.2104 ± 0.0019	0.2090 ± 0.0018	0.2119 ± 0.0018	0.2043 ± 0.0023	0.2932 ± 0.0020	0.2219 ± 0.0024
poker-hand	0.4987 ± 0.0006	0.4987 ± 0.0006	0.4987 ± 0.0006	0.4987 ± 0.0006	0.4987 ± 0.0006	0.4987 ± 0.0006	0.5392 ± 0.0006	0.4987 ± 0.0005
donation	0.0081 ± 0.0009	0.0081 ± 0.0009	0.0081 ± 0.0009	0.0081 ± 0.0009	0.0081 ± 0.0009	0.0081 ± 0.0009	0.0120 ± 0.0005	0.0082 ± 0.0009

References

1. Duda, R.O.; Hart, P.E. *Pattern Classification and Scene Analysis*; John Wiley and Sons: Hoboken, NJ, USA, 1973.
2. Zaidi, N.A.; Carman, M.J.; Cerquides, J.; Webb, G.I. Naive-bayes inspired effective pre-conditioner for speeding-up logistic regression. In Proceedings of the IEEE International Conference on Data Mining, Shenzhen, China, 14–17 December 2014; pp. 1097–1102.
3. Domingos, P.; Pazzani, M. Beyond independence: Conditions for the optimality of the simple bayesian classifier. In Proceedings of the 13th International Conference on Machine Learning, Bari, Italy, 3–6 July 1996; pp. 105–112.
4. Friedman, N.; Geiger, D.; Goldszmidt, M. Bayesian network classifiers. *Mach. Learn.* **1997**, *29*, 131–163. [CrossRef]
5. Webb, G.I.; Boughton, J.R.; Wang, Z. Not so naive bayes: Aggregating one-dependence estimators. *Mach. Learn.* **2005**, *58*, 5–24. [CrossRef]
6. Sahami, M. Learning limited dependence bayesian classifiers. In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, Portland, OR, USA, 2–4 August 1996; ACM: New York, NY, USA, 1996; pp. 335–338.
7. Wang, L.; Liu, Y.; Mammadov, M.; Sun, M.; Qi, S. Discriminative structure learning of bayesian network classifiers from training dataset and testing instance. *Entropy* **2019**, *21*, 489. [CrossRef] [PubMed]
8. Jiang, L.; Zhang, H.; Cai, Z.; Su, J. Learning tree augmented naive bayes for ranking. In *Database Systems for Advanced Applications*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 688–698.
9. Alhussan, A.; El Hindi, K. Selectively fine-tuning bayesian network learning algorithm. *Int. J. Pattern Recognit. Artif. Intell.* **2016**, *30*, 1651005. [CrossRef]
10. Wang, Z.; Webb, G.I.; Zheng, F. Adjusting dependence relations for semi-lazy tan classifiers. In *AI 2003: Advances in Artificial Intelligence*; Gedeon, T.D., Fung, L.C.C., Eds.; Springer: Berlin/Heidelberg, Germany, 2003; pp. 453–465.
11. De Campos, C.P.; Corani, G.; Scanagatta, M.; Cuccu, M.; Zaffalon, M. Learning extended tree augmented naive structures. *Int. J. Approx. Reason.* **2016**, *68*, 153–163. [CrossRef]
12. Jiang, L.; Cai, Z.; Wang, D.; Zhang, H. Improving tree augmented naive bayes for class probability estimation. *Knowl.-Based Syst.* **2012**, *26*, 239–245. [CrossRef]
13. Cerquides, J.; De Mántaras, R.L. TAN classifiers based on decomposable distributions. *Mach. Learn.* **2005**, *59*, 323–354. [CrossRef]
14. Zhang, L.; Jiang, L.; Li, C. A discriminative model selection approach and its application to text classification. *Neural Comput. Appl.* **2019**, *31*, 1173–1187. [CrossRef]
15. Langley, P.; Sage, S. Induction of selective bayesian classifiers. In Proceedings of the 10th International Conference on Uncertainty in Artificial Intelligence, Nice, France, 21–23 September 2016; Morgan Kaufmann Publishers Inc., MIT: Cambridge, MA, USA, 1994; pp. 399–406.
16. Zheng, F.; Webb, G.I. Finding the right family: Parent and child selection for averaged one-dependence estimators. In *European Conference on Machine Learning*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 490–501.
17. Chen, S.; Webb, G.I.; Liu, L.; Ma, X. A novel selective naive bayes algorithm. *Knowl.-Based Syst.* **2020**, *192*, 105361. [CrossRef]
18. Martínez, A.M.; Webb, G.I.; Chen, S.; Zaidi, N.A. Scalable learning of bayesian network classifiers. *J. Mach. Learn. Res.* **2016**, *17*, 1–35.
19. Chen, S.; Martínez, A.M.; Webb, G.I. Highly scalable attributes selection for averaged one-dependence estimators. In Proceedings of the 18th Pacific-Asia Conference on Knowledge Discovery and Data Mining, Tainan, Taiwan, 13–16 May 2014; Springer: Berlin/Heidelberg, Germany, 2014; pp. 86–97.
20. Chen, S.; Martínez, A.M.; Webb, G.I.; Wang, L. Sample-based attribute selective ande for large data. *IEEE Trans. Knowl. Data Eng.* **2017**, *29*, 172–185. [CrossRef]
21. Zaidi, N.A.; Webb, G.I.; Carman, M.J.; Petitjean, F.; Buntine, W.; Hynes, M.; Sterck, H.D. Efficient parameter learning of bayesian network classifiers. *Mach. Learn.* **2017**, *106*, 1289–1329. [CrossRef]
22. Brown, G.; Pocock, A.; Zhao, M.J.; Lujan, M. Conditional likelihood maximisation: A unifying framework for information theoretic feature selection. *J. Mach. Learn. Res.* **2012**, *13*, 27–66.
23. Yu, L.; Liu, H. Feature selection for high-dimensional data: A fast correlation-based filter solution. In Proceedings of the Twentieth International Conference on International Conference on Machine Learning, ICML'03, Washington, DC, USA 21–24 August 2003; pp. 856–863.
24. Peng, H.; Long, F.; Ding, C. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1226–1238. [CrossRef] [PubMed]
25. Battiti, R. Using mutual information for selecting features in supervised neural net learning. *IEEE Trans. Neural Netw.* **1994**, *5*, 537–550. [CrossRef] [PubMed]
26. Fleuret, F. Fast binary feature selection with conditional mutual information. *J. Mach. Learn. Res.* **2004**, *5*, 1531–1555.
27. Chen, S.; Martínez, A.M.; Webb, G.I.; Wang, L. Selective A*n*DE for large data learning: A low-bias memory constrained approach. *Knowl. Inf. Syst.* **2017**, *50*, 475–503. [CrossRef]
28. Data visualization and feature selection: New algorithms for nongaussian data. *Adv. Neural Inf. Process. Syst.* **2000**, *12*, 687–693.
29. PMeyer, E.; Schretter, C.; Bontempi, G. Information-theoretic feature selection in microarray data using variable complementarity. *IEEE J. Sel. Top. Signal Process.* **2008**, *2*, 261–274.

30. Kohavi, R. The power of decision tables. In *European Conference on Machine Learning*; Lavrac, N., Wrobel, S., Eds.; Springer: Berlin/Heidelberg, Germany, 1995; pp. 174–189.
31. Dua, D.; Graff, C. UCI Machine Learning Repository. 2017. Available online: http://archive.ics.uci.edu/ml (accessed on 5 September 2021).
32. Witten, I.H.; Frank, E.; Trigg, L.E.; Hall, M.A.; Holmes, G.; Cunningham, S.J. Weka: Practical Machine Learning Tools and Techniques with JAVA Implementations. Available online: https://researchcommons.waikato.ac.nz/handle/10289/1040 (accessed on 9 October 2021).
33. Cestnik, B. Estimating probabilities: A crucial task in machine learning. In Proceedings of the European Conference on Artificial Intelligence, Stockholm, Sweden, 1 January 1990; Volume 90, pp. 147–149.
34. Flores, M.J.; Gámez, J.A.; Martínez, A.M.; Puerta, J.M. Handling numeric attributes when comparing bayesian network classifiers: does the discretization method matter? *Appl. Intell.* **2011**, *34*, 372–385. [CrossRef]
35. Demšar, J. Statistical comparisons of classifiers over multiple data sets. *J. Mach. Learn. Res.* **2006**, *7*, 1–30.