Expert Systems with Applications 42 (2015) 5433–5440

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Not always simple classification: Learning SuperParent for class probability estimation

Chen Qiu [a], Liangxiao Jiang [a,*], Chaoqun Li [b]

[a] Department of Computer Science, China University of Geosciences, Wuhan 430074, China
[b] Department of Mathematics, China University of Geosciences, Wuhan 430074, China

ARTICLE INFO

Article history:
Available online 5 March 2015

Keywords:
SuperParent
CLL-SuperParent
Conditional log likelihood
Classification accuracy
AUC

ABSTRACT

Of numerous proposals to improve naive Bayes (NB) by weakening its attribute independence assumption, SuperParent (SP) has demonstrated remarkable classification performance. In many real-world applications, however, accurate class probability estimation of instances is more desirable than simple classification. For example, we often need to recommend commodities to customers with the higher likelihood (class probability) of purchase. Conditional log likelihood (CLL) is currently a well-accepted measure for the quality of class probability estimation. Inspired by this, in this paper, we firstly investigate the class probability estimation performance of SP in terms of CLL and find that its class probability estimation performance almost ties the original distribution-based tree augmented naive Bayes (TAN). In order to scale up its class probability estimation performance, we then propose an improved CLL-based SuperParent algorithm (CLL-SP). In CLL-SP, a CLL-based approach, instead of a classification-based approach, is used to find the augmenting arcs. The experimental results on a large suite of benchmark datasets show that our CLL-based approach (CLL-SP) significantly outperforms the classification-based approach (SP) and the original distribution-based approach (TAN) in terms of CLL, yet at the same time maintains the high classification accuracy that characterizes the classification-based approach (SP).

1. Introduction

Classification is a basic task in data mining and machine learning. The goal of learning algorithms for classification is to construct a classifier given instances with class labels. The classifier predicts the possible class label to instances described by a set of attribute values. The predictive ability of a classifier is typically measured by its classification accuracy or error rate on testing instances. To the best of our knowledge, these classifiers can be broadly divided into two main categories: probability-based classifiers and margin-based classifiers. In this paper, we focus our attention on probability-based classifiers and find that probability-based classifiers can also produce probability estimates and offer "confidence" of the class prediction-that is, the information how "far-off" (be it 0.99 or 0.01?) is the prediction of each instance from its true class label. Yet, this information is often ignored when we use these probability-based classifiers for simple classification.

In probability-based classification, accurate class probability estimation plays the most important role. In fact, accurate class probability estimation is also widely used in other paradigms of machine learning, such as probability-based ranking (Provost & Domingos, 2003) and cost-sensitive learning (Margineantu, 2005; Wang, Qin, Zhang, & Zhang, 2012), and many other paradigms, such as information retrieval (Gupta, Saini, & Saxena, 2015), distance learning (Li & Li, 2013), expert and intelligent systems (Bohanec & Rajkovic, 1988; Kurgan, Cios, Tadeusiewicz, Ogiela, & Goodenday, 2001), and recommendation systems (Bobadilla, Serradilla, & Bernal, 2010). For example (Saar-Tsechansky & Provost, 2004), in target making, the estimated probability that a customer will respond to an offer is combined with the estimated profit to evaluate various offer propositions. For another example, in cost-sensitive decision-making, the class probability estimation is used to minimize the conditional risk (Domingos, 1999; Elkan, 2001). Besides, the estimated class membership probabilities are often used for ranking of cases (Saar-Tsechansky & Provost, 2004; Provost & Domingos, 2003), to improve response rate to provide different service strategies for different users. For instance, these statistics can help us to recommend the customer products of higher purchasing probability, which means, it may interest the customer and be profitable to the seller. These examples raise the following question: Can we directly learn a probability-based

* Corresponding author. Tel./fax: +86 27 67883716.
E-mail addresses: qiuchen1114@gmail.com (C. Qiu), ljiang@cug.edu.cn (L. Jiang), chqli@cug.edu.cn (C. Li).

http://dx.doi.org/10.1016/j.eswa.2015.02.049
0957-4174/

classifier for optimizing its class probability estimate performance?

To answer this question, we firstly need a proper measure to evaluate a classifier in terms of its class probability estimation performance. It is conditional log likelihood (Grossman & Domingos, 2004; Guo & Greiner, 2005; Jiang, Zhang, & Cai, 2009; Jiang, Cai, & Wang, 2012), denoted by CLL. CLL is currently a well-accepted measure for the quality of class probability estimation. Given a classifier G and a set of test instances $T = \{e_1, e_2, \ldots, e_i \ldots, e_t\}$, where t is the number of test instances. Let c_i be the true class label of e_i. Then, the conditional log likelihood $CLL(G|T)$ of the classifier G on the test instance set T is defined as:

$$CLL(G|T) = \sum_{i=1}^{t} logP_G(c_i|e_i), \tag{1}$$

where $P_G(c_i|e_i)$ is the probability that the classifier G predicts the test instance e_i belonging to its true class c_i.

Let e, represented by an attribute value vector $< a_1, a_2, \ldots, a_m >$, be an arbitrary test instance and the true class label of it be c, then we can use the built classifier G to estimate the probability that e belongs to c. Now, the only left question to answer is how to estimate the class membership probability $P(c|e)$ using the constructed probability-based classifiers. Bayesian network classifiers are typical probability-based classifiers, which estimate $P(c|e)$ using Eq. (2).

$$\begin{aligned} P(c|e) &= P(c|a_1, a_2, \ldots, a_m) = \frac{P(c)P(a_1, a_2, \ldots, a_m|c)}{P(a_1, a_2, \ldots, a_m)} \\ &= \frac{P(c)P(a_1, a_2, \ldots, a_m|c)}{\sum_c P(c)P(a_1, a_2, \ldots, a_m|c)}. \end{aligned} \tag{2}$$

Assume that all attributes are fully independent given the class. Then, the resulting Bayesian network classifier is called naive Bayes (NB). NB estimates $P(c|e)$ using Eq. (3).

$$P(c|e) = \frac{P(c)\prod_{j=1}^{m} P(a_j|c)}{\sum_c P(c)\prod_{j=1}^{m} P(a_j|c)}, \tag{3}$$

where the prior probability $P(c)$ with Laplace correction is defined by Eq. (4), and the conditional probability $P(a_j|c)$ with Laplace correction is defined by Eq. (5).

$$P(c) = \frac{\sum_{i=1}^{n} \delta(c_i, c) + 1}{n + n_c}, \tag{4}$$

$$P(a_j|c) = \frac{\sum_{i=1}^{n} \delta(a_{ij}, a_j)\delta(c_i, c) + 1}{\sum_{i=1}^{n} \delta(c_i, c) + n_j}, \tag{5}$$

where n is the number of training instances, n_c is the number of classes, n_j is the number of values of the jth attribute, c_i is the class label of the ith training instance, a_{ij} is the jth attribute value of the ith training instance, a_j is the jth attribute value of the test instance, and $\delta(\bullet)$ is a binary function, which is one if its two parameters are identical and zero otherwise. Thus, $\sum_{i=1}^{n} \delta(c_i, c)$ is the frequency that the class label c occurs in the training data and $\sum_{i=1}^{n} \delta(a_{ij}, a_j)\delta(c_i, c)$ is the frequency that the class label c and the attribute value a_j occur simultaneously in the training data.

Fig. 1 shows graphically an example of naive Bayes (NB). In NB, each attribute node has the class node as its parent, but does not have any parent from other attribute nodes. Because the values of $P(c)$ and $P(a_j|c)$ can be easily estimated from training instances, NB is easy to construct. The structure of NB is the simplest form of Bayesian networks. It is obvious that the conditional independence assumption in NB is rarely true in reality, which would harm its performance in the applications with complex attribute dependencies. Since attribute dependencies can be explicitly represented by

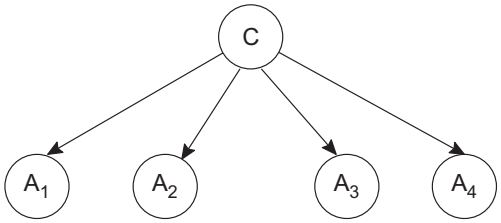

Fig. 1. An example of NB.

arcs, extending the structure of naive Bayes is a direct way to overcome its limitation. For example, tree augmented naive Bayes (TAN) (Friedman, Geiger, & Goldszmidt, 1997) is an extended tree-like naive Bayes, in which the class node directly points to all attribute nodes and an attribute node has at most one parent from another attribute node. Fig. 2 shows graphically an example of TAN.

To learn the structure of TAN, a distribution-based approach (Friedman et al., 1997) is originally proposed and has demonstrated remarkable classification performance. In order to scale up its classification performance, a classification-based approach, simply called SuperParent (SP), is proposed by Keogh and Pazzani (1999). Although SP has already been proved to be an effective classification algorithm, its class probability estimation performance, in terms of conditional log likelihood (CLL), is unknown. Inspired by this, in this paper, we firstly investigate the class probability estimation performance of SP in terms of CLL and find that its class probability estimation performance almost ties the original distribution-based tree augmented naive Bayes (TAN). To scale up its class probability estimation performance, we then propose an improved CLL-based SuperParent algorithm (CLL-SP). In CLL-SP, a CLL-based approach, instead of a classification-based approach, is used to find the augmenting arcs.

The rest of this paper is organized as follows. In Section 2, we simply introduce some related works on improving naive Bayes, especially revisit the SuperParent algorithm (SP). In Section 3, we propose an improved CLL-based SuperParent algorithm (CLL-SP) for class probability estimation. In Section 4, we conduct a series of experiments on a large suite of benchmark datasets to validate our proposed algorithm. In Section 5, we draw conclusions and outline the main directions for our future work.

2. Related work

Naive Bayes (NB) has emerged as a simple and effective classification algorithm for data mining and machine learning, but its conditional independence assumption is violated in many real-world applications. Intuitively, the Bayesian networks can provide a powerful model for arbitrary attribute dependencies (Pearl, 1988). Unfortunately, it has been proved that learning an optimal Bayesian network is NP-hard (Non-deterministic Polynomial-time hard) (Chickering, 1996). In order to avoid the intractable complexity for learning Bayesian networks, learning improved naive Bayes

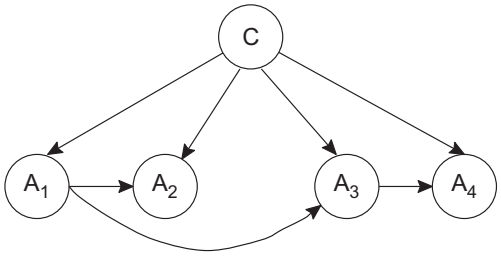

Fig. 2. An example of TAN.

have attracted much attention from researchers, and the related approaches can be broadly divided into five main categories (Jiang, 2011): (1) structure extension; (2) attribute weighting; (3) attribute selection; (4) instance weighting; (5) instance selection, also called local learning.

The approach of structure extension attempts to augment the structure of naive Bayes and uses directed arcs to explicitly represent attribute dependencies. For example, tree augmented naive Bayes (TAN) (Friedman et al., 1997) is an extended tree-like naive Bayes, in which the class node directly points to all attribute nodes and an attribute node has at most one parent node from another attribute node, and thus TAN estimates $P(c|e)$ using Eq. (6).

$$P(c|e) = \frac{P(c)\prod_{j=1}^{m} P(a_j|c, a_{jp})}{\sum_c P(c)\prod_{j=1}^{m} P(a_j|c, a_{jp})}, \tag{6}$$

where a_{jp} is the attribute value of the parent node A_{jp} of the attribute node A_j. For example, in Fig. 2, A_1 is the parent node of the attribute nodes A_2 and A_3, and A_3 is the parent node of the attribute node A_4. Please note that, when $A_{jp} = \varnothing$, the attribute node A_j does not have parent node from any other attribute nodes, and thus $P(a_j|c, a_{jp})$ is simplified into $P(a_j|c)$.

Seen from Eq. (6), a key step of learning a TAN is to find the parent node of each attribute node, and thus structure learning is unavoidable in TAN. To learn the structure of TAN, a distribution-based approach (Friedman et al., 1997) is originally proposed and has demonstrated remarkable classification performance. This distribution-based approach (Friedman et al., 1997) computes the conditional mutual information between each pair of attributes, and then builds a complete maximum weighted spanning tree among attributes to directly approximate the underlying probability distribution.

However, according to the conclusions drawn by Keogh and Pazzani (1999), directly approximating the underlying probability distribution is not always the best way to improve classification accuracy. In order to further improve its classification performance, a classification-based approach, simply called SuperParent (SP), is proposed. SuperParent (SP) (Keogh & Pazzani, 1999) conducts a forward hill climbing search to find the best arc, the arc which improves classification accuracy of the current classifier at most, to add on each iteration. Please note that, SP breaks this up into two steps, first finding a good parent and then finding the best child of that parent. The compared results by Keogh and Pazzani (1999) show that SP is significantly better than TAN indeed.

Aiming at accurate ranking, Ling and Zhang (2002) propose an AUC-based approach, and thus the resulting algorithm is called AUC-SuperParent (AUC-SP). Different from SP, AUC-SP uses AUC (the area under the receiver operating characteristics curve, the area under the ROC curve) (Bradley & Andrew, 1997; Hand & Till, 2001) as the score function to find the augmenting arcs. This kind of discriminative learning approach (Grossman & Domingos, 2004; Guo & Greiner, 2005; Jiang, Zhang, & Cai, 2006) perfectly matches the learning process (maximizing AUC) and the learning goal of ranking, and thus the learned classifiers can produce higher AUC values. Since AUC is currently a well-accepted measure for the quality of ranking, AUC-SuperParent (AUC-SP) is often used for ranking.

3. CLL-SuperParent

Numerous algorithms have been proposed to improve naive Bayes (NB) by weakening its conditional attribute independence assumption, among which tree augmented naive Bayes (TAN) has demonstrated remarkable classification performance in terms of classification accuracy or error rate. A key step of learning a TAN is to find the parent node of each attribute node, namely how to learn its structure is crucial.

Although SP achieves significant improvement on classification, its class probability estimation performance, in terms of conditional log likelihood (CLL) (Grossman & Domingos, 2004; Guo & Greiner, 2005; Jiang et al., 2009, 2012), is unknown. Thus it is interesting whether SP can also achieve significant improvement on class probability estimation, in terms of CLL. In order to answer this question, we conduct a group of experiments to empirically investigate its class probability estimation performance in terms of CLL. Unfortunately, our experimental results show that SP almost ties the original distribution-based tree augmented naive Bayes (TAN). To our knowledge, this is attributable to a mismatch between the learning process (maximizing classification accuracy) and the learning goal of class probability estimation (maximizing conditional log likelihood).

Inspired by this, in this paper, we would like to focus our attention to improve SuperParent (SP) for class probability estimation, since accurate class probability estimation is more desirable than just classification in many real-world applications. For example, we often need to deploy different promotion strategies to customers with different likelihood (class probability) of buying some products. We call our improved algorithm CLL-based SuperParent (CLL-SP). In CLL-SP, a CLL-based approach, instead of a classification-based approach, is used to find the augmenting arcs, and thus our proposed CLL-SP perfectly matches the learning process (maximizing conditional log likelihood) and the learning goal of class probability estimation (maximizing conditional log likelihood). Now, let's describe the outline of our proposed CLL-SP as follows:

Table 1
Descriptions of 36 UCI datasets used in our experiments.

Dataset	Inst.	Num.	Nom.	Class	% Missing
Anneal	898	6	32	5	0.0
Anneal.ORIG	898	6	32	5	63.3
Audiology	226	0	69	24	2.0
Autos	205	15	10	7	1.1
Balance-scale	625	4	0	3	0.0
Breast-cancer	286	0	9	2	0.3
Breast-w	699	9	0	2	0.3
Colic	368	7	15	2	23.8
Colic.ORIG	368	7	20	2	18.7
Credit-a	690	6	9	2	0.6
Credit-g	1000	7	13	2	0.0
Diabetes	768	8	0	2	0.0
Glass	214	9	0	7	0.0
Heart-c	303	6	7	5	0.2
Heart-h	294	6	7	5	20.4
Heart-statlog	270	13	0	2	0.0
Hepatitis	155	6	13	2	5.6
Hypothyroid	3772	23	6	4	6.0
Ionosphere	351	34	0	2	0.0
Iris	150	4	0	3	0.0
kr-vs-kp	3196	0	36	2	0.0
Labor	57	8	8	2	3.9
Letter	20000	16	0	26	0.0
Lymph	148	3	15	4	0.0
Mushroom	8124	0	22	2	1.4
Primary-tumor	339	0	17	21	3.9
Segment	2310	19	0	7	0.0
Sick	3772	7	22	2	6.0
Sonar	208	60	0	2	0.0
Soybean	683	0	35	19	9.8
Splice	3190	0	61	3	0.0
Vehicle	846	18	0	4	0.0
Vote	435	0	16	2	5.6
Vowel	990	10	3	11	0.0
Waveform-5000	5000	40	0	3	0.0
Zoo	101	1	16	7	0.0

Algorithm 1. An outline of CLL-SuperParent (CLL-SP)

1. Initiation: Initialize network to naive Bayes.
2. Evaluation: Evaluate the current classifier in terms of its CLL.
3. Finding SuperParent: Consider making each node a SuperParent. Let A_{sp} be the SuperParent which increases CLL the most.
4. Finding FavoriteChild: Consider an arc from A_{sp} to each orphan. If the best such arc improves CLL, then keep it and go to 3; Else stop the search process and return the current classifier.

Seen from above outline, our proposed CLL-SP initializes network to naive Bayes and then evaluates its class probability estimation performance in terms of conditional log likelihood (CLL). Thirdly, the effect on conditional log likelihood (CLL), of making each node a SuperParent is assessed, and the best such node is chosen as the SuperParent denoted by A_{sp}. Finally, CLL-SP finds FavoriteChild of A_{sp}, by assessing the effect of adding a single arc from A_{sp} to each orphan. If the addition of the arc from A_{sp} to the FavoriteChild improves the conditional log likelihood (CLL), the arc is added into the current classifier, and the SuperParent cycle begins again. If there was no improvement on the conditional log likelihood (CLL), the current classifier is returned.

To some extent, CLL-SP is a variant of the original SP for class probability estimation. Different from SP, CLL-SP uses conditional log likelihood (CLL), instead of classification accuracy, as the objective function to find each SuperParent and its FavoriteChild, and then to find the appropriate structure of TAN with high class probability estimation performance. The experimental results on a large suite of benchmark datasets validate the effectiveness of CLL-SP.

4. Experiments and results

In this section, we design a group of experiments to empirically investigate the class probability estimation performance of SuperParent (SP) and to validate the effectiveness of our proposed CLL-based SuperParent (CLL-SP). So, we compare related algorithms such as CLL-SP, NB, TAN, SP, and AUC-SP in terms of conditional log likelihood (CLL) defined by Eq. (1). Please note that, in our implementations, the Laplace correction is used to smooth the related probability estimates in all of the compared algorithms.

We ran our experiments on 36 benchmark UCI (University of California, Irvine) datasets published on the main web site of WEKA platform (Witten, Frank, & Hall, 2011), which represent a wide range of domains and data characteristics listed in Table 1. Please note that, the missing values(%) column shows the percentage of a dataset's entries (number of features × number of instances) that have missing values (Hall, 2000; Frank, Hall, & Pfahringer, 2003). In our experiments, missing values are replaced with the modes and means from the available data. Numeric attribute values are discretized using the unsupervised ten-bin

Table 2
Comparison of conditional log likelihood for CLL-SP versus NB, TAN, SP, and AUC-SP.

Dataset	CLL-SP	NB	TAN	SP	AUC-SP
Anneal.ORIG	−30.5	−34.0 •	−34.7 •	−32.3	−29.1
Anneal	−7.6	−20.5 •	−16.0 •	−10.9	−8.2
Audiology	−73.2	−95.1 •	−119.5 •	−94.7 •	−84.2 •
Autos	−29.7	−65.6 •	−51.0 •	−45.6 •	−30.4
Balance-scale	−48.1	−45.8 ○	−48.5	−45.8 ○	−46.1
Breast-cancer	−27.4	−26.5	−27.1	−27.2	−29.0
Breast-w	−12.7	−26.4 •	−13.9	−23.8 •	−13.3
Colic.ORIG	−29.5	−30.6	−50.8 •	−30.1	−30.5
Colic	−34.6	−44.2 •	−35.3	−41.7 •	−39.5 •
Credit-a	−37.9	−41.5 •	−44.0 •	−41.4 •	−39.9
Credit-g	−80.2	−76.2	−84.8	−77.1	−81.2
Diabetes	−60.5	−58.8	−58.9	−59.0	−60.2
Glass	−35.8	−34.7	−39.3	−35.2	−35.0
Heart-c	−19.5	−20.1	−21.5	−20.3	−21.8
Heart-h	−18.9	−19.5	−20.3	−19.6	−19.7
Heart-statlog	−17.9	−17.7	−19.4	−17.8	−19.1
Hepatitis	−10.0	−12.3	−11.0	−12.6	−10.6
Hypothyroid	−130.8	−140.2 •	−138.3 •	−137.2 •	−134.7 •
Ionosphere	−28.5	−50.2 •	−27.4	−49.8 •	−33.3
Iris	−3.0	−3.7	−3.5	−3.4	−3.2
kr-vs-kp	−55.0	−134.9 •	−81.9 •	−80.2 •	−69.3 •
Labor	−1.8	−1.0	−3.5	−1.1	−1.2
Letter	−1765.9	−3614.2 •	−1851.4 •	−1773.2	−1755.8
Lymph	−9.8	−9.0	−10.8	−8.9	−9.3
Mushroom	−0.1	−152.6 •	−0.4	−5.2	−3.4 •
Primary-tumor	−94.6	−94.6	−100.6 •	−94.8	−94.9
Segment	−66.0	−179.4 •	−70.5	−76.6	−62.2
Sick	−31.5	−66.4 •	−42.0 •	−42.1 •	−33.4
Sonar	−29.5	−32.7	−31.6	−32.0	−30.6
Soybean	−13.3	−37.9 •	−13.6	−32.1 •	−23.0 •
Splice	−65.7	−67.1	−70.9	−65.2	−66.2
Vehicle	−88.8	−248.3 •	−87.0	−122.3 •	−101.8 •
Vote	−11.0	−39.3 •	−11.2	−23.9 •	−11.8
Vowel	−32.2	−129.5 •	−36.6 •	−37.4 •	−34.5
Waveform-5000	−329.7	−545.3 •	−349.0 •	−348.3	−356.3 •
Zoo	−1.1	−1.8 •	−1.5	−1.6	−1.6
W/T/L	–	1/15/20	0/23/13	1/22/13	0/28/8

○, • statistically significant improvement or degradation over CLL-SP.

discretization implemented in WEKA platform. Besides, we manually delete three useless attributes: the attribute "Hospital Number" in the data set "colic.ORIG", the attribute "instance name" in the data set "splice", and the attribute "animal" in the data set "zoo".

In our experiments, the conditional log likelihood (CLL) of each algorithm on each dataset are obtained via 10 runs of 10-fold cross-validation. We use the paired t-test (Nadeau & Bengio, 2003) for comparison of two classifiers and the Friedman test with the corresponding post hoc tests (Alcalá-Fdez et al., 2009; Demasal, 2006) such as Bergmann test for comparison of more classifiers over multiple data sets.

4.1. Paired t-test

Table 2 shows the detailed compared results in terms of conditional log likelihood (CLL). The symbols ∘ and • in the table respectively denote statistically significant improvement or degradation over our proposed CLL-SP with the $p = 0.05$ significance level (Nadeau & Bengio, 2003). In other words, The symbol • means that our proposed CLL-SP is statistically better than its competitors and the symbol ∘ means that our proposed CLL-SP is statistically worse than its competitors. Besides, the $W/T/L$ values on 36 datasets are summarized at the bottom of the table. Each entry $W/T/L$ in the table means that NB, TAN, SP, and AUC-SP win on W datasets, tie on T datasets, and lose on L datasets, compared to our proposed CLL-SP. Finally, we conducted a corrected paired two-tailed t-test with the $p = 0.05$ significance level (Nadeau & Bengio, 2003) to compare each pair of algorithms. Tables 3 and 4 respectively show the summary and ranking test results. In Table 3, for each entry $i(j), i$ is the number of datasets on which the algorithm in the column achieves higher CLL than the algorithm in the corresponding row, and j is the number of datasets on which the algorithm in the column achieves significant wins with the $p = 0.05$ significance level (Nadeau & Bengio, 2003) with regard to the algorithm in the corresponding row. In Table 4, the first column is the difference between the total number of wins and the total number of losses that the corresponding algorithm achieves compared with all the other algorithms, which is used to generate the ranking. The second and third columns represent the total numbers of wins and losses respectively. From our experimental results, we can see that the classification-based approach (SP) almost ties the original distribution-based approach (TAN), and our proposed CLL-based approach (CLL-SP) significantly outperforms all the other compared approaches in terms of conditional log likelihood (CLL). Now, we summarize the highlights as follows:

1. In terms of CLL, seen from Table 3, SP almost ties TAN (7 wins and 5 losses).
2. In terms of CLL, seen from Table 2, CLL-SP significantly outperforms all the other algorithms such as NB (20 wins and 1 loss), TAN (13 wins and 0 loss), SP (13 wins and 1 loss), and AUC-SP (8 wins and 0 loss).
3. In terms of CLL, seen from Table 4, the overall ranking (in descending order) among all of the compared algorithms are CLL-SP (54 wins and 2 losses), AUC-SP (36 wins and 14 losses), SP (21 wins and 28 losses), TAN (21 wins and 35 losses), and NB (9 wins and 62 losses), respectively.

4.2. Friedman test

Table 5 shows average rankings of the algorithms obtained by applying the Friedman test (Alcalá-Fdez et al., 2009; Demasal, 2006). With five algorithms and 36 data sets, F_F is distributed according to the F distribution with $5-1=4$ and $(5-1)\times(36-1)=140$ degrees of freedom. F_F calculated from the mean ranks is 11.0284, which is greater than the critical value of $F(4,140)$ for $a = 0.05$. So we reject the null-hypothesis and proceed with a Bergmann test to find out exactly the significant difference among these algorithms. Table 6 reports the z-values and the p-values obtained, where the detailed results using the Bergmann test indicates which algorithms are significantly different. From the test results, we can see that:

1. The average ranking of CLL-SP (1.8056) is much higher than those of AUC-SP (2.7778), SP (3.0694), TAN (3.6111) and NB (3.7361).

Table 3
Compared results of the corrected paired two-tailed t-test ($p = 0.05$): conditional log likelihood.

	NB	TAN	SP	AUC-SP	CLL-SP
NB	–	20 (13)	25 (12)	25 (17)	28 (20)
TAN	16 (6)	–	21 (7)	25 (9)	32 (13)
SP	10 (0)	15 (5)	–	22 (10)	27 (13)
AUC-SP	11 (2)	11 (3)	14 (1)	–	28 (8)
CLL-SP	8 (1)	4 (0)	9 (1)	8 (0)	–

Table 4
Compared results of ranking tests: conditional log likelihood.

Resultset	Wins–losses	Wins	Losses
CLL-SP	52	54	2
AUC-SP	22	36	14
SP	−7	21	28
TAN	−14	21	35
NB	−53	9	62

Table 5
Average rankings of the algorithms obtained by applying the Friedman test: conditional log likelihood.

Algorithm	Ranking
CLL-SP	1.8056
NB	3.7361
TAN	3.6111
SP	3.0694
AUC-SP	2.7778

Table 6
P-values for $\alpha = 0.05$: conditional log likelihood.

i	Algorithms	$z = (R_0 - R_i)/SE$	p
10	NB vs. CLL-SP	5.180224	0
9	TAN vs. CLL-SP	4.844814	0.000001
8	SP vs. CLL-SP	3.39137	0.000695
7	AUC-SP vs. CLL-SP	2.608746	0.009087
6	NB vs. AUC-SP	2.571478	0.010127
5	TAN vs. AUC-SP	2.236068	0.025347
4	NB vs. SP	1.788854	0.073638
3	TAN vs. SP	1.453444	0.1461
2	SP vs. AUC-SP	0.782624	0.433848
1	NB vs. TAN	0.33541	0.737316

Bergmann's procedure rejects these hypotheses:

- NB vs. CLL-SP
- TAN vs. CLL-SP
- SP vs. CLL-SP
- AUC-SP vs. CLL-SP

2. According to the Bergmann test, the class probability estimation performance of CLL-SP is significantly better than those of NB, TAN, SP, amd AUC-SP.

To further prove the advantages and applications of our proposed CLL-based approach (CLL-SP), we observe its performance on another two datasets: the cardiac diagnosis dataset (Kurgan et al., 2001) and the car evaluation dataset (Bohanec & Rajkovic, 1988). The cardiac diagnosis dataset describes diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT) images and the car evaluation dataset is used for testing constructive induction and structure discovery methods. Figs. 3 and 4 graphically show the detailed results comparing CLL. From these experimental results, we can see that our proposed CLL-based SuperParent (CLL-SP) is also much better than SuperParent (SP) and AUC-based SuperParent (AUC-SP) in terms of conditional log likelihood (CLL).

Finally, we also observe its classification and ranking performance in terms of classification accuracy and the area under the ROC curve (AUC). In our experiments, the classification accuracy is the percentage of test instances correctly classified, and the AUC of a classifier on a data set with two classes is computed by Eq. (7).

$$\hat{A} = \frac{S_0 - n_0 \times (n_0 + 1)/2}{n_0 \times n_1}, \tag{7}$$

where n_0 and n_1 are the numbers of negative and positive instances respectively, and $S_0 = \sum r_i$, where r_i is the rank of the i_{th} negative instance in the ranked list.

For multiple classes, AUC can be calculated using the following M measure (Hand & Till, 2001).

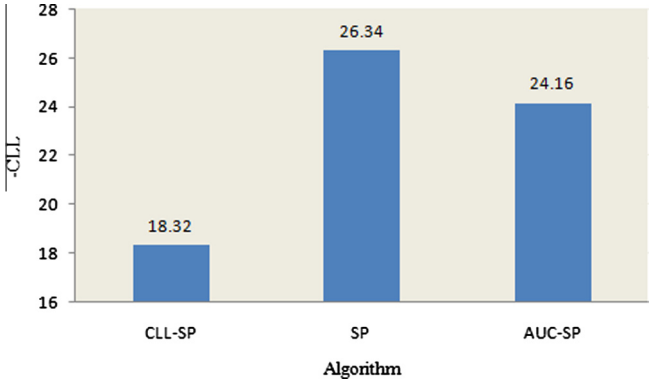

Fig. 3. Negative CLL comparisons for CLL-SP versus SP and AUC-SP on the cardiac diagnosis dataset.

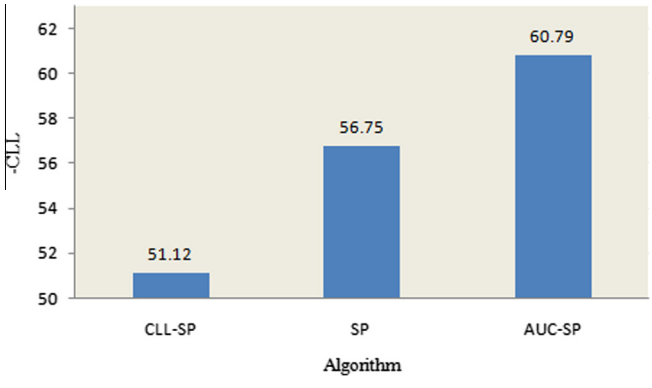

Fig. 4. Negative CLL comparisons for CLL-SP versus SP and AUC-SP on the car evaluation dataset.

Table 7
Comparison of classification accuracy (mean$\times 10^2$) for CLL-SP versus NB, TAN, SP, and AUC-SP.

Dataset	CLL-SP	NB	TAN	SP	AUC-SP
Anneal.ORIG	89.7	88.2 •	90.5	89.7	91.0
Anneal	98.7	94.3 •	96.7 •	98.2	98.0
Audiology	72.6	71.4	65.3 •	71.2	72.1
Autos	80.6	64.0 •	72.5 •	72.3 •	80.7
Balance-scale	89.9	91.4 ○	86.1 •	91.4 ○	91.2
Breast-cancer	69.9	72.9	69.5	72.0	70.9
Breast-w	96.5	97.3	95.5	97.3	97.0
Colic.ORIG	75.2	74.2	67.7 •	74.3	75.5
Colic	80.4	78.9	80.1	80.3	80.1
Credit-a	85.2	84.7	84.1	84.6	85.1
Credit-g	75.2	75.9	74.9	75.4	75.2
Diabetes	75.7	75.7	76.3	75.9	76.7
Glass	56.4	57.7	58.7	57.9	60.8
Heart-c	82.5	83.4	79.7	82.8	82.3
Heart-h	82.6	83.6	81.3	83.5	83.5
Heart-statlog	82.5	83.8	79.5	83.9	82.2
Hepatitis	85.1	84.1	83.0	83.9	86.6
Hypothyroid	93.3	92.8 •	93.4	93.0	93.2
Ionosphere	92.3	90.9	91.3	90.8	92.3
Iris	94.9	94.3	94.3	94.5	94.5
kr-vs-kp	96.3	87.8 •	92.9 •	94.8 •	96.3
Labor	93.9	96.7	89.0	96.9	95.4
Letter	82.9	70.1 •	82.7	83.4	83.2
Lymph	86.6	86.0	83.7	85.6	86.2
Mushroom	100.0	95.5 •	100.0	99.9	99.9 •
Primary-tumor	47.4	47.2	44.8	46.7	47.3
Segment	93.2	89.0 •	93.9	93.2	93.9
Sick	97.9	96.8 •	97.7	97.7	97.7
Sonar	77.6	76.3	75.3	76.4	78.3
Soybean	94.6	92.2 •	95.0	92.9	93.9
Splice	95.5	95.4	94.9	95.6	95.4
Vehicle	71.8	61.0 •	73.3	69.9	72.9
Vote	94.7	90.2 •	94.4	92.2 •	95.1
Vowel	92.2	66.1 •	91.9	91.2	91.6
Waveform-5000	82.5	80.0 •	80.4 •	82.4	82.3
Zoo	96.9	94.4	96.6	94.4	94.5
W/T/L	–	1/21/14	0/29/7	1/32/3	0/35/1

○, • statistically significant improvement or degradation over CLL-SP.

Table 8
Compared results of the corrected paired two-tailed t-test ($p = 0.05$): classification accuracy.

	NB	TAN	SP	AUC-SP	CLL-SP
NB	–	19 (13)	23 (12)	27 (14)	27 (14)
TAN	17 (6)	–	23 (10)	30 (7)	29 (7)
SP	12 (0)	13 (3)	–	22 (3)	22 (3)
AUC-SP	9 (0)	6 (1)	14 (0)	–	18 (1)
CLL-SP	9 (1)	7 (0)	14 (1)	18 (0)	–

Table 9
Compared results of ranking tests: classification accuracy.

Resultset	Wins–Losses	Wins	Losses
CLL-SP	23	25	2
AUC-SP	22	24	2
SP	14	23	9
TAN	−13	17	30
NB	−46	7	53

Table 10
Average rankings of the algorithms obtained by applying the Friedman test: classification accuracy.

Algorithm	Ranking
CLL-SP	2.3333
NB	3.6944
TAN	3.75
SP	2.9444
AUC-SP	2.2778

$$\hat{A} = \frac{2}{n_c(n_c-1)} \sum_{i<j} \hat{A}(c_i, c_j), \tag{8}$$

where n_c is the number of classes and the $\hat{A}(c_i, c_j)$ term is the AUC value of each pair of classes c_i and c_j.

Table 11
P-values for $\alpha = 0.05$: classification accuracy.

i	Algorithms	$z = (R_0 - R_i)/SE$	p
10	TAN vs. AUC-SP	3.950387	0.000078
9	TAN vs. CLL-SP	3.801316	0.000144
8	NB vs. AUC-SP	3.801316	0.000144
7	NB vs. CLL-SP	3.652244	0.00026
6	TAN vs. SP	2.161532	0.030654
5	NB vs. SP	2.012461	0.044171
4	SP vs. AUC-SP	1.788854	0.073638
3	SP vs. CLL-SP	1.639783	0.10105
2	NB vs. TAN	0.149071	0.881497
1	AUC-SP vs. CLL-SP	0.149071	0.881497

Bergmann's procedure rejects these hypotheses:

- NB vs. AUC-SP
- NB vs. CLL-SP
- TAN vs. AUC-SP
- TAN vs. CLL-SP

Table 12
Comparison of area under the ROC curve (mean $\times 10^2$) for CLL-SP versus NB, TAN, SP, and AUC-SP.

Dataset	CLL-SP	NB	TAN	SP	AUC-SP
Anneal.ORIG	96.5	95.4 •	96.8	96.1	97.1 ○
Anneal	99.7	98.9 •	99.6	99.4	99.7
Audiology	97.8	96.9 •	95.9 •	96.9 •	97.3 •
Autos	95.4	88.1 •	92.7 •	92.3 •	95.6
Balance-scale	95.2	96.2 ○	94.0 •	96.2 ○	96.0
Breast-cancer	65.4	70.2	65.4	69.6	67.8
Breast-w	99.2	99.2	98.9	99.3	99.3
Colic.ORIG	82.3	81.2	72.0 •	82.2	83.0
Colic	85.9	84.4	85.6	84.8	84.7
Credit-a	92.2	91.9	90.5 •	91.8	92.2
Credit-g	78.0	79.1	77.1	78.9	77.8
Diabetes	81.3	82.6	81.9	82.9	82.7
Glass	79.9	80.2	80.7	80.9	82.9 ○
Heart-c	90.3	91.1	87.9	91.0	90.0
Heart-h	89.7	90.0	87.1	89.9	90.0
Heart-statlog	90.7	91.3	89.4	91.1	90.5
Hepatitis	89.1	89.4	87.1	88.9	90.2
Hypothyroid	85.2	83.4 •	83.2 •	83.8 •	85.1
Ionosphere	96.4	93.7 •	98.0 ○	93.8 •	96.6
Iris	99.1	99.0	99.0	99.0	99.0
kr-vs-kp	99.5	95.2 •	98.3 •	98.8 •	99.5
Labor	95.8	98.7	93.7	98.2	98.7
Letter	99.2	97.1 •	99.1 •	99.2	99.3
Lymph	92.8	92.9	92.0	93.1	93.2
Mushroom	100.0	99.8 •	100.0	100.0	100.0
Primary-tumor	83.4	82.8	81.0 •	82.6	83.3
Segment	99.4	98.4 •	99.5 ○	99.4	99.5
Sick	99.0	95.9 •	98.1 •	98.2 •	99.2
Sonar	87.0	85.5	83.8	85.6	86.8
Soybean	99.7	99.5 •	99.8	99.6 •	99.7
Splice	99.4	99.4	99.2 •	99.4	99.4
Vehicle	90.9	83.5 •	91.8	90.0	91.4
Vote	98.6	97.1 •	98.7	97.9	98.7
Vowel	99.6	95.8 •	99.5 •	99.5	99.6
Waveform-5000	95.5	95.3	94.5 •	95.7	95.8 ○
Zoo	100.0	99.9	99.9	99.9	99.9
W/T/L	–	1/20/15	2/21/13	1/28/7	3/32/1

○, • statistically significant improvement or degradation over CLL-SP.

The detailed compared results are shown in Tables 7–16. From these experimental results, we can see that our proposed CLL-SP maintains the high classification accuracy that characterizes the classification-based approach (SP) and almost ties the AUC-based approach (AUC-SP) in terms of AUC. Some highlights are summarized as:

1. In terms of classification accuracy, CLL-SP almost ties SP (3 wins and 1 loss) and AUC-SP (1 win and 0 loss).
2. In terms of AUC, CLL-SP slightly outperforms SP (7 wins and 1 loss) and almost ties AUC-SP (1 win and 3 losses).
3. According to the Friedman test with a Bergmann post hoc test, the hypotheses CLL-SP vs. SP and CLL-SP vs. AUC-SP are accepted.

Table 13
Compared results of the corrected paired two-tailed t-test ($p = 0.05$): area under the ROC curve.

	NB	TAN	SP	AUC-SP	CLL-SP
NB	–	15 (13)	24 (13)	29 (17)	24 (15)
TAN	21 (8)	–	25 (9)	29 (13)	28 (13)
SP	10 (0)	11 (3)	–	27 (8)	23 (7)
AUC-SP	6 (0)	7 (0)	9 (0)	–	14 (1)
CLL-SP	12 (1)	8 (2)	13 (1)	22 (3)	–

Table 14
Compared results of ranking tests: area under the ROC curve.

Resultset	Wins–losses	Wins	Losses
AUC-SP	40	41	1
CLL-SP	29	36	7
SP	5	23	18
TAN	−25	18	43
NB	−49	9	58

Table 15
Average rankings of the algorithms obtained by applying the Friedman test: area under the ROC curve.

Algorithm	Ranking
CLL-SP	2.7083
NB	3.5694
TAN	3.5972
SP	2.8611
AUC-SP	2.2639

Table 16
P-values for $\alpha = 0.05$: area under the ROC curve.

i	Algorithms	$z = (R_0 - R_i)/SE$	p
10	TAN vs. AUC-SP	3.577709	0.000347
9	NB vs. AUC-SP	3.503173	0.00046
8	TAN vs. CLL-SP	2.385139	0.017073
7	NB vs. CLL-SP	2.310604	0.020855
6	TAN vs. SP	1.975193	0.048246
5	NB vs. SP	1.900658	0.057347
4	SP vs. AUC-SP	1.602515	0.109042
3	AUC-SP vs. CLL-SP	1.19257	0.233038
2	SP vs. CLL-SP	0.409946	0.681846
1	NB vs. TAN	0.074536	0.940584

Bergmann's procedure rejects these hypotheses:

- NB vs. AUC-SP
- TAN vs. AUC-SP

5. Conclusion and future work

In many real-world applications, such as intelligent medical diagnostic systems, recommendation systems and cost-sensitive decision-making systems, accuracy class probability estimation of instances are more desirable than simple classification. In this paper, we firstly investigate the class probability estimation performance of the state-of-the-art SuperParent (SP) and then propose an improved CLL-based SuperParent algorithm (CLL-SP). In CLL-SP, a CLL-based approach, instead of a classification-based approach, is used to find the augmenting arcs. The experimental results on a large number of UCI datasets validate the effectiveness of our proposed CLL-SP in terms of conditional log likelihood (CLL), classification accuracy, and the area under the ROC curve (AUC).

Although the experimental results impressive, we would like to discuss the possible limitations of the proposed algorithm as follows. Firstly, the current version of CLL-SP uses a single objective metric (conditional log likelihood) to evaluate the augmenting arcs and the resulted classifiers. Secondly, CLL-SP does not take the misclassification costs into account and thus is a cost-insensitive learning algorithm. Thirdly, our current work handles nominal attributes only and numeric attribute values are discretized using the unsupervised ten-bin discretization.

In the future, the main research directions can be concluded as: (1) We plan to study further some other objective metrics like predictive information and cross entropy. Besides, no proposals with hybrid multi-objective metrics have been explored so far, which could be challenging in terms of selecting the best next arc. (2) Adapting CLL-SP to cost-sensitive scenario is an interesting future issue. (3) Extending it to deal with numeric attributes is also interesting. (4) Using CLL-SP to enhance the accuracy of probability-based distance metrics is another topic for future research. (5) Exploring further its applications to some real-world expert and intelligent systems, such as intelligent medical diagnostic systems, recommendation systems and cost-sensitive decision-making systems, should be considered again.

Acknowledgments

The work was partially supported by the National Natural Science Foundation of China (61203287), the Program for New Century Excellent Talents in University (NCET-12-0953), the Chenguang Program of Science and Technology of Wuhan (201550431073), and the Fundamental Research Funds for the Central Universities (CUG130504, CUG130414).

References

Alcalá-Fdez, J., Sánchez, L., García, S., Jesus, M. J. d., Ventura, S. J., Garrell, M., et al. (2009). Keel: A software tool to assess evolutionary algorithms to data mining problems. *Soft Computing, 13*(3), 307–318.

Bobadilla, J., Serradilla, F., & Bernal, J. (2010). A new collaborative filtering metric that improves the behavior of recommender systems. *Knowledge-Based Systems, 23*(6), 520–528.

Bohanec, M., & Rajkovic, V. (1988). Knowledge acquisition and explanation for multi-attribute decision making. In *Proceedings of the eighth intl workshop on expert systems and their applications, Avignon, France* (pp. 59–78).

Bradley, P., & Andrew, P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition, 30*(7), 1145–1159.

Chickering, D. M. (1996). Learning bayesian networks is np-complete. In D. Fisher & H. Lenz (Eds.), *Learning from data: Artificial intelligence and statistics V* (pp. 121–130). Springer-Verlag.

Demasal, J. (2006). Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research, 7*, 1–30.

Domingos, P. (1999). Metacost: A general method for making classifiers cost-sensitive. In *Proceedings of the fifth ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 155–164). ACM.

Elkan, C. (2001). The foundations of cost-sensitive learning. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence* (pp. 973–978). Lawrence Erlbaum Associates Ltd.

Frank, E., Hall, M., & Pfahringer, B. (2003). Locally weighted naive bayes. In *Proceedings of the conference on uncertainty in artificial intelligence* (pp. 249–256). Morgan Kaufmann.

Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning, 29*(2–3), 131–163.

Grossman, D., & Domingos, P. (2004). Learning bayesian network classifiers by maximizing conditional likelihood. In *Proceedings of the twenty-first international conference on machine learning* (pp. 361–368). ACM Press.

Guo, Y., & Greiner, R. (2005). Discriminative model selection for belief net structures. In *Proceedings of the twentieth national conference on artificial intelligence* (pp. 770–776). AAAI Press.

Gupta, Y., Saini, A., & Saxena, A. K. (2015). A new fuzzy logic based ranking function for efficient information retrieval system. *Expert Systems with Applications, 42*(3), 1223–1234.

Hall, M. (2000). Correlation-based feature selection for discrete and numeric class machine learning. In *Proceedings of the 17th international conference on machine learning* (pp. 359–366).

Hand, D., & Till, R. (2001). A simple generalisation of the area under the ROC curve for multiple class classification problems. *Machine Learning, 45*(2), 171–186.

Jiang, L. (2011). Random one-dependence estimators. *Pattern Recognition Letters, 32*(3), 532–539.

Jiang, L., Cai, Z., & Wang, D. (2012). Improving tree augmented naive Bayes for class probability estimation. *Knowledge-Based Systems, 26*, 239–245.

Jiang, L., Zhang, H., & Cai, Z. (2006). Discriminatively improving naive Bayes by evolutionary feature selection. *Romanian Journal of Information Science and Technology, 9*(3), 163–174.

Jiang, L., Zhang, H., & Cai, Z. (2009). A novel Bayes model: Hidden naive bayes. *IEEE Transactions on Knowledge and Data Engineering, 21*(10), 1361–1371.

Keogh, E., & Pazzani, M. (1999). 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* (pp. 225–230).

Kurgan, L. A., Cios, K. J., Tadeusiewicz, R., Ogiela, M., & Goodenday, L. S. (2001). Knowledge discovery approach to automated cardiac spect diagnosis. *Artificial Intelligence in Medicine, 23*, 149.

Li, C., & Li, H. (2013). Bayesian network classifiers for probability-based metrics. *Journal of Experimental & Theoretical Artificial Intelligence, 25*(4), 477–491.

Ling, C., & Zhang, H. (2002). Toward Bayesian classifiers with accurate probabilities. In *Advances in knowledge discovery and data mining* (pp. 123–134). Berlin Heidelberg: Springer.

Margineantu, D. D. (2005). Active cost-sensitive learning. In *Proceedings of the 19th international conference on artificial intelligence* (pp. 1622–1623).

Nadeau, C., & Bengio, Y. (2003). Inference for the generalization error. *Machine Learning, 52*, 239–281.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems*. San Francisco: Morgan Kaufmann.

Provost, F., & Domingos, P. (2003). Tree induction for probability-based ranking. *Machine Learning, 52*(3), 199–215.

Saar-Tsechansky, M., & Provost, F. (2004). Active sampling for class probability estimation and ranking. *Machine Learning, 54*(2), 153–178.

Wang, T., Qin, Z., Zhang, S., & Zhang, C. (2012). Cost-sensitive classification with inadequate labeled data. *Information Systems, 37*, 508–516.

Witten, I., Frank, E., & Hall, M. (2011). *Data mining: Practical machine learning tools and techniques* (3rd ed.). Morgan Kaufmann.