Journal of Experimental & Theoretical Artificial Intelligence

ISSN: 0952-813X (Print) 1362-3079 (Online) Journal homepage: http://www.tandfonline.com/loi/teta20

Naive Bayes for optimal ranking

Harry Zhang & Jiang Su

To cite this article: Harry Zhang & Jiang Su (2008) Naive Bayes for optimal ranking, Journal of Experimental & Theoretical Artificial Intelligence, 20:2, 79-93, DOI: 10.1080/09528130701476391

To link to this article: https://doi.org/10.1080/09528130701476391

Published online: 08 May 2008.

Journal of Experimental & Theoretical Artificial Intelligence,
Vol. 20, No. 2, June 2008, 79–93

Taylor & Francis
Taylor & Francis Group

Naive Bayes for optimal ranking†

HARRY ZHANG* and JIANG SU

Faculty of Computer Science, University of
New Brunswick, New Brunswick E3B 5A3, Canada

(*Received 24 August 2006; in final form 25 May 2007*)

It is well known that naive Bayes performs surprisingly well in classification, but its probability estimation is poor. AUC (the area under the receiver operating characteristics curve) is a measure different from classification accuracy and probability estimation, which is often used to measure the quality of rankings. Indeed, an accurate ranking of examples is often more desirable than a mere classification. What is the general performance of naive Bayes in yielding optimal ranking, measured by AUC? In this paper, we study it systematically by both empirical experiments and theoretical analysis. In our experiments, we compare naive Bayes with a state-of-the-art decision-tree learning algorithm C4.4 for ranking, and some popular extensions of naive Bayes which achieve a significant improvement over naive Bayes in classification, such as the selective Bayesian classifier (SBC) and tree-augmented naive Bayes (TAN). Our experimental results show that naive Bayes performs significantly better than C4.4 and comparably with TAN. This provides empirical evidence that naive Bayes performs well in ranking. Then we analyse theoretically the optimality of naive Bayes in ranking. We study two example problems: conjunctive concepts and *m*-of-*n* concepts, which have been used in analysing the performance of naive Bayes in classification. Surprisingly, naive Bayes performs optimally on them in ranking, even though it does not in classification. We present and prove a sufficient condition for the optimality of naive Bayes in ranking. From both empirical and theoretical studies, we believe that naive Bayes is a competitive model for ranking.

Keywords: Naive Bayes; Ranking; Classification; AUC; ROC curve

*Corresponding author. Email: hzhang@unb.ca
†A preliminary version of this paper appeared in ECML2004

Journal of Experimental & Theoretical Artificial Intelligence
ISSN 0952–813X print/ISSN 1362–3079 online
http://www.tandf.co.uk/journals
DOI: 10.1080/09528130701476391

1. Introduction

Naive Bayes is an effective and efficient classification algorithm. In classification learning problems, a learner attempts to construct a classifier from a given set of training examples with class labels. An example E is represented by a vector of attribute-values $(a_1, a_2, \ldots, a_n)$, where a_i is the value of attribute A_i. Let C represent the class variable. We use c to represent the value taken by C.

There are numerous classification algorithms, such as decision trees, Bayesian networks, and neural networks. From the probabilistic point of view, a probability distribution $p(A_1, \ldots, A_n, C)$ is learned from the training data, and an example E is classified into the class c with the maximum posterior class probability $p(c|E)$ (or simply class probability) as shown below:

$$C_b(E) = \arg\max_c p(c|E). \tag{1}$$

$C_b(E)$ is called a Bayesian classifier.

Assume that all attributes are independent given the value of the class variable (conditional independence assumption), i.e.

$$p(E|c) = p(a_1, a_2, \ldots, a_n|c) = \prod_{i=1}^{n} p(a_i|c). \tag{2}$$

The resulting classifier is then

$$C_{nb}(E) = \arg\max_c p(c) \prod_{i=1}^{n} p(a_i|c). \tag{3}$$

$C_{nb}(E)$ is called a naive Bayesian classifier, or simply naive Bayes (NB). Figure 1 shows an example of naive Bayes. In naive Bayes, each attribute node has the same class node as its parent, but does not have any parent from attribute nodes.

Because the values of $p(a_i|c)$ can be estimated from training examples, naive Bayes is easy to construct. However, it is also surprisingly effective (Kononenko 1990, Langley *et al*. 1992, Domingos and Pazzani 1997). Naive Bayes is based on the conditional independence assumption that all attributes are independent given the class. It is obvious that the conditional independence assumption is rarely true in reality. Indeed, naive Bayes is found to work poorly for regression problems (Frank *et al*. 2000) and produce poor probability estimates (Bennett 2000).

Typically, the performance of a classifier is measured by its predictive accuracy (or error rate). Some classifiers, such as naive Bayes and decision trees, also produce estimates of the class probability $p(c|E)$. This information is often ignored in classification, as long as the class with the highest class probability estimate is

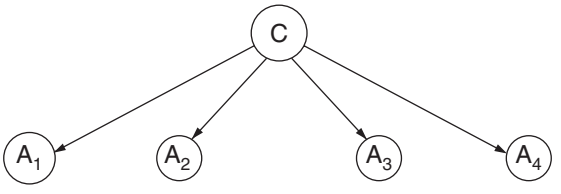

Figure 1. An example of naive Bayes.

identical to the actual class. However, in many applications, classification and error rate are not enough. For example, a computer science department needs a ranking of its students in terms of their performance in various aspects in order to award scholarships. Thus a ranking is desired. If a ranking is desired and only a data set with class labels is given, the area under the receiver operating characteristics (ROC) curve (Swets 1988, Provost and Fawett 1997), or simply AUC can be used to evaluate the quality of rankings generated by a classifier. AUC is a good 'summary' for comparing two classifiers across the entire range of class distributions and error costs. Bradley (1997) shows that AUC is a proper metric for the quality of classifiers averaged across all possible probability thresholds. It has been shown that, for binary classification, AUC is equivalent to the probability that a randomly chosen example of class $-$ will have a smaller estimated probability of belonging to class $+$ than a randomly chosen example of class $+$ (Hand and Till 2001). The AUC of a classifier can be computed using the following formula:

$$\hat{A} = \frac{S_0 - n_0(n_0+1)/2}{n_0 n_1} \tag{4}$$

where n_0 and n_1 are the numbers of negative and positive examples, respectively, and $S_0 = \sum r_i$, where r_i is the rank of the ith positive example in the ranking. It is clear from equation (4) that AUC is essentially a measure of the quality of a ranking. For example, the AUC of a ranking is 1 (the maximum value of AUC) if there is no positive example preceding a negative example. Some researchers believe that AUC is a better and more discriminating evaluation method than accuracy for classifiers which produce class probability estimates (Ling *et al*. 2003a).

It is surprising that naive Bayes performs well in classification, because the conditional independence assumption that it is based on almost never holds. Domingos and Pazzani (1997) present an explanation which ascribes the good classification performance of naive Bayes to the zero-one loss function. This function defines the error as the number of incorrect classifications (Friedman 1996). Unlike other loss functions, such as the squared error, the zero–one loss function does not penalize inaccurate probability estimation as long as the maximum probability is assigned to the correct class. This means that naive Bayes may change the posterior probabilities of each class, but the class with the maximum posterior probability is often unchanged. Thus the classification is still correct, although the probability estimation is poor. For example, let us assume that the true probabilities $p(+|E)$ and $p(-|E)$ are 0.9 and 0.1, respectively, and that the probability estimates $\hat{p}(+|E)$ and $\hat{p}(-|E)$ produced by naive Bayes are 0.6 and 0.4. Obviously, the probability estimates are poor, but the classification (positive) is not affected. This means that naive Bayes tolerates the estimation error of class probabilities to some extent.

It is interesting to notice that a similar story happens in ranking. Note that the ranking addressed in this paper is based on the class probabilities of examples. Ranking is different from both classification and probability estimation. For example, assume that E_+ and E_- are a positive and a negative example, respectively, and that the actual class probabilities are $p(+|E_+) = 0.9$ and $p(+|E_-) = 0.1$. An algorithm which gives class probability estimates $\hat{p}(+|E_+) = 0.55$ and $\hat{p}(+|E_-) = 0.54$ gives a correct order of E_+ and E_- in the ranking. Notice that the probability estimates are poor and the classification for E_- is incorrect (assume that the threshold for classification is 0.5).

Since naive Bayes works well in classification, a natural question is: what is the ranking performance of naive Bayes, measured by AUC? Does naive Bayes perform in ranking as well as in classification? The main motivation of this paper is to answer these questions empirically and theoretically.

The rest of the paper is organized as follows. Section 2 reviews the related work. Section 3 describes an empirical study showing that naive Bayes performs well in generating accurate rankings, which provides empirical evidence that naive Bayes has good performance in ranking. Section 4 explores the theoretical reason for the superb performance of naive Bayes in ranking. The paper concludes with a summary and discussion of our work.

2. Related work

Naive Bayes is easy to construct and has surprisingly good performance in classification, even though the conditional independence assumption is rarely true in real world applications. Numerous techniques have been proposed to improve or extend naive Bayes. One approach is to select a subset of attributes in which attributes are conditionally independent. The idea of selecting a subset of attributes or forming new attributes is to convert the data to a new form which satisfies the conditional independence assumption. Of the proposed techniques, the *selective Bayesian classifier* (SBC) (Langley and Sage 1994) demonstrates a remarkable improvement over naive Bayes in classification. SBC uses forward selection to find a good subset of attributes, and then uses this subset to construct a naive Bayes.

A more effective and straightforward way to alleviate the conditional independence assumption of naive Bayes is to extend its structure to represent explicitly attribute dependencies by adding arcs between attributes. Tree augmented naive Bayes (TAN) is an extended tree-like naive Bayes (Friedman *et al*. 1997) in which the class node directly points to all attribute nodes and an attribute node can have only one parent from another attribute node. Figure 2 shows an example of TAN.

Friedman *et al*. (1997) propose a TAN learning algorithm, called CL-TAN in this paper, based on conditional mutual information, defined as follows.

$$I_P(X; Y|Z) = \sum_{x,y,z} P(x, y, z)\log\frac{P(x, y|z)}{P(x|z)P(y|z)} \tag{5}$$

where x, y, and z are the values of variables X, Y, and Z, respectively. In CL-TAN, $I_P(A_i; A_j|C)$ between each pair of attributes is computed, and a complete undirected

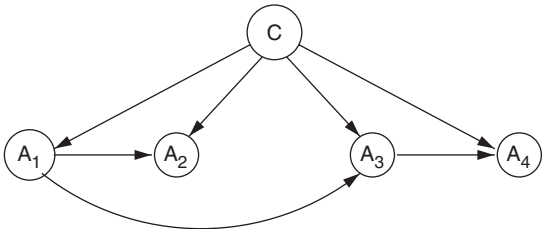

Figure 2. An example of TAN.

weighted graph is built, in which nodes are attributes $A_1, \ldots, A_n$, and the weight of an edge connecting A_i to A_j is set to $I_P(A_i; A_j|C)$. Then, a maximum weighted spanning tree is constructed. Finally, the undirected tree is converted to the directed one, and a node labelled with C which points to all attribute nodes is added.

However, most extensions of naive Bayes aim at improving the predictive accuracy, not at better probability estimation or ranking. Actually, naive Bayes is found to produce poor probability estimates (Bennett 2000). Some work has been proposed to improve the probability estimates yielded by a learning algorithm, called probability calibration. Platt scaling (Platt 1999) and isotonic regression (Zadrozny and Elkan 2001, 2002) are the two major methods used for probability calibration. Isotonic regression has been used successfully to calibrate the probability estimates of naive Bayes (Zadrozny and Elkan 2001, 2002). An extensive empirical study in comparing Platt scaling and isotonic regression has been performed recently (Niculescu and Caruana 2005), and shows that Platt scaling also achieves good effect in calibrating the probability estimates of naive Bayes. However, probability calibration aims at accurate probability estimates, different from ranking.

Decision tree learning algorithms are one of the simplest and most effective learning algorithms, widely used in many applications. Traditional decision-tree learning algorithms, such as C4.5 (Quinlan 1993), are error based, and also produce probability estimates. Unfortunately, traditional decision-tree algorithms, such as C4.5, have been observed to produce poor probability estimates (Pazzani *et al*. 1994, Provost *et al*. 1998). Provost and Domingos (2003) propose two techniques to improve the AUC of C4.5.

(i) **Smooth probability estimates by Laplace correction**. Assume that there are p examples of the class c at a leaf, N total examples, and C total classes. The frequency-based estimation calculates the estimated probability $p(c)$ as p/N. The Laplace estimation calculates the estimated probability $p(c)$ as $(\mathrm{p}+1)/(\mathrm{N}+\mathrm{C})$.

(ii) **Turn off pruning**. Provost and Domingos (2003) show that pruning a large tree damages the probability estimation. Thus a simple strategy to improve the probability estimation is to build a large tree without pruning.

The resulting algorithm is called C4.4. C4.4 achieves a significant improvement over C4.5 with regard to AUC. Ling and Yan (2003) propose a method to calibrate the probability estimate generated by C4.5. Ferri *et al*. (2002), present a novel algorithm for learning decision trees, which is based on AUC, rather than entropy.

To our knowledge, there is no systematic study of the performance of naive Bayes with respect to ranking, measured by AUC.

3. Empirical study

We conducted experiments to investigate the ranking performance of naive Bayes. Our aim is to compare naive Bayes with its extensions which demonstrate remarkable improvement in classification, such as SBC and TAN. From our experiments, we observed that CL-TAN is sensitive to the edge directions, and removing the edges with weak dependencies could improve the performance in AUC (Jiang *et al*. 2005). Thus we modify the CL-TAN algorithm as follows.

(i) The attribute with the maximum mutual information with class $I_P(A;C)$, defined by

$$I_P(A;C) = \sum_{A,C} P(A,C) \log \frac{P(A,C)}{P(A)P(C)} \tag{6}$$

is chosen as the root, and the directions of all edges are set to be outward from it.

(ii) The edges with conditional mutual information less than the average conditional mutual information I_{avg}, defined by

$$I_{\text{avg}} = \frac{\sum_i \sum_{j,j\neq i} I_P(A_i; A_j|C)}{n(n-1)} \tag{7}$$

where n is the number of attributes, are removed.

Since the structure of the resulting model is a forest instead of a tree, we call our algorithm FAN (Jiang *et al*. 2005). FAN shows a considerable improvement over CL-TAN in AUC.

We conducted experiments on the 36 UCI data sets (Merz *et al*. 1997) recommended by Witten and Frank (2000), which represent a wide range of domains and data characteristics. The description of the data sets is given in table 1. We downloaded these data sets in the *arff* format from the main web of Weka. In our experiments, we adopted the following three preprocessing steps on each data set.

(i) Missing values. We used the unsupervised filter *Replace Missing Values* in Weka to replace the missing values in each data set.

(ii) Discretization of numeric attributes. We used the unsupervised filter *Discretize* in Weka to discretize the numerical values of attributes in each data set.

(iii) Removal of useless attributes. Apparently, if the number of values of an attribute is almost equal to the number of examples in a data set, it does not contribute useful information to future prediction. For example, student ID numbers are useless for prediction. Thus we removed this type of attributes using the unsupervised filter *Remove* in Weka. In fact, only the three attributes named 'Hospital Number' in data set *colic.ORIG*, 'instance name' in data set *splice*, and 'animal' in data set *zoo* were deleted.

We compared naive Bayes with the selective Bayesian classifier (SBC) (Langley and Sage 1994), TAN (Friedman *et al*. 1997), FAN, and C4.4 (Provost and Domingos 2003) in AUC. All algorithms were implemented within the Weka framework. Multi-class AUC was calculated by M-measure (Hand and Till 2001). The AUC of each classifier was obtained via 10 runs of 10-fold cross validation on a data set. Runs with the various classifiers were carried out on the same training sets and evaluated on the same test sets. Throughout, we compared each pair of algorithms via a two-tailed t-test with a significantly different probability of 0.95.

Table 2 shows the AUC and standard deviations of each classifier on each data set, and the average AUC and deviation are summarized at the bottom of the table. Table 3 shows the results of the two-tailed t-test between each pair of algorithms, and each entry $w/t/l$ means that the algorithm in the corresponding row wins in w data

Table 1. Description of data sets used in the experiments.

No.	Date set	Examples	Attributes	Classes	Missing	Numeric
1	Anneal	898	39	6	Y	Y
2	Anneal.ORIG	898	39	6	Y	Y
3	Audiology	226	70	24	Y	N
4	Autos	205	26	7	Y	Y
5	Balance-scale	625	5	3	N	Y
6	Breast-cancer	286	10	2	Y	N
7	Breast-w	699	10	2	Y	N
8	Colic	368	23	2	Y	Y
9	Colic.ORIG	368	28	2	Y	Y
10	Credit-a	690	16	2	Y	Y
11	Credit-g	1000	21	2	N	Y
12	Diabetes	768	9	2	N	Y
13	Glass	214	10	7	N	Y
14	Heart-c	303	14	5	Y	Y
15	Heart-h	294	14	5	Y	Y
16	Heart-statlog	270	14	2	N	Y
17	Hepatitis	155	20	2	Y	Y
18	Hypothyroid	3772	30	4	Y	Y
19	Ionosphere	351	35	2	N	Y
20	Iris	150	5	3	N	Y
21	kr-vs-kp	3196	37	2	N	N
22	Labor	57	17	2	Y	Y
23	Letter	20,000	17	26	N	Y
24	Lymphography	148	19	4	N	Y
25	Mushroom	8124	23	2	Y	N
26	Primary-tumor	339	18	21	Y	N
27	Segment	2310	20	7	N	Y
28	Sick	3772	30	2	Y	Y
29	Sonar	208	61	2	N	Y
30	Soybean	683	36	19	Y	N
31	Splice	3190	62	3	N	N
32	Vehicle	846	19	4	N	Y
33	Vvote	435	17	2	Y	N
34	Vowel	990	14	11	N	Y
35	Waveform-5000	5000	41	3	N	Y
36	Zoo	101	18	7	N	Y

sets, ties in t data sets, and loses in l data sets compared with the algorithm in the corresponding column. We summarize the highlights as follows.

(i) Naive Bayes outperforms C4.4 significantly: It wins in 23 data sets, ties in eight data sets, and loses in five data sets. The average AUC for naive Bayes is 89.50%, higher than the average AUC 85.71% of C4.4. Notice that C4.4 is the state-of-the-art decision-tree algorithm designed for yielding accurate rankings.

(ii) Naive Bayes is comparable with CL-TAN (seven wins and nine losses) and FAN (eight wins and ten losses), and outperforms SBC (nine wins and five losses). Notice that both SBC and CL-TAN have achieved a significant improvement over naive Bayes in classification. However, their improvement in AUC is not clear.

Table 2. Experimental results on AUC.

Date sets	C4.4	NB	SBC	CL-TAN	FAN
Anneal	94.42 ± 2.36	96.1 ± 1.18	95.12 ± 2.38	96.56 ± 0.21	96.51 ± 0.36
Anneal.ORIG	92.27 ± 2.97	94.26 ± 4.23	94.27 ± 4.35	95.1 ± 2.89	95.02 ± 3.18
Audiology	70.5 ± 0.73	71.08 ± 0.64	70.92 ± 0.74	70.95 ± 0.6	70.76 ± 0.6
Autos	91.1 ± 3.58	90.07 ± 4.93	91.08 ± 4.14	92.45 ± 4.34	92.77 ± 4.29
Balance-scale	61.89 ± 6.88	84.08 ± 4.42	84.08 ± 4.42	77.31 ± 5.44	81.07 ± 6.25
Breast-cancer	58.47 ± 10.14	68.24 ± 11.93	67.43 ± 12.7	61.54 ± 10.57	61.51 ± 11.47
Breast-w	98.08 ± 1.29	99.22 ± 0.76	99.2 ± 0.76	98.71 ± 1.07	99.04 ± 0.88
Colic	82.58 ± 7.95	83.22 ± 7.46	84.54 ± 6.75	84.59 ± 6.56	85.08 ± 6.33
Colic.ORIG	81.92 ± 6.73	80.57 ± 8.03	80.46 ± 7.83	73.02 ± 9.77	72.99 ± 9.87
Credit-a	89.18 ± 3.87	91.71 ± 3.16	86.78 ± 4.76	89.57 ± 3.68	90.51 ± 3.43
Credit-g	69.52 ± 4.8	79.02 ± 4.22	77.72 ± 4.92	76.45 ± 5.08	77.32 ± 5.08
Diabetes	75.84 ± 5.1	82.51 ± 5	82.17 ± 6.4	80.54 ± 4.96	81.63 ± 5.02
Glass	82.73 ± 4.68	80.89 ± 5.9	81.17 ± 5.95	83.03 ± 6.42	83.18 ± 6.69
Heart-c	83.14 ± 0.77	84.05 ± 0.6	83.77 ± 0.67	83.56 ± 0.76	83.6 ± 0.73
Heart-h	83.19 ± 0.67	83.9 ± 0.62	83.23 ± 0.94	83.45 ± 0.69	83.55 ± 0.66
Heart-statlog	80.99 ± 8.93	90.85 ± 5.12	87.63 ± 7.03	87.48 ± 5.24	87.49 ± 5.46
Hepatitis	76.89 ± 15.78	88.41 ± 10.91	81.96 ± 14.28	84.67 ± 11.01	85.49 ± 10.47
Hypothyroid	81.25 ± 7.62	87.78 ± 6.12	85.43 ± 5.61	87.25 ± 6.87	87.84 ± 6.68
Ionosphere	93 ± 4.58	93.4 ± 4.79	93.06 ± 5.33	98.09 ± 2.17	98.14 ± 2.23
Iris	96.83 ± 2.83	98.64 ± 2.17	98.43 ± 2	98.52 ± 2.46	98.45 ± 2.4
kr-vs-kp	99.93 ± 0.08	95.16 ± 1.2	96.35 ± 0.9	98.22 ± 0.56	98.2 ± 0.57
Labor	78 ± 21.83	98.17 ± 7.36	73.21 ± 25.02	88.75 ± 17.02	90.92 ± 14.8
Letter	95.4 ± 0.32	96.88 ± 0.21	97.04 ± 0.2	98.91 ± 0.11	98.81 ± 0.13
Lymph	86.43 ± 4.6	90 ± 1.71	87.99 ± 3.55	89.3 ± 3.05	89.82 ± 2.49
Mmushroom	100 ± 0	99.79 ± 0.07	99.98 ± 0.02	100 ± 0	100 ± 0
Primary-tumor	75.22 ± 2.24	78.88 ± 1.76	78.28 ± 1.89	78.3 ± 1.81	78.21 ± 1.79
Segment	98.9 ± 0.41	98.5 ± 0.41	98.94 ± 0.36	99.53 ± 0.22	99.58 ± 0.21
Sick	98.97 ± 0.7	95.83 ± 2.4	94.75 ± 3.39	98.31 ± 1.01	98.35 ± 0.97
Sonar	76.65 ± 9.03	84.17 ± 9.52	75.32 ± 11.54	79.84 ± 10.5	81.24 ± 9.88
Soybean	91.23 ± 1.55	99.73 ± 0.34	98.81 ± 1.16	99.71 ± 0.41	99.75 ± 0.4
Splice	98.09 ± 0.66	99.45 ± 0.28	99.2 ± 0.42	99.37 ± 0.35	99.37 ± 0.35
Vehicle	85.57 ± 2.89	80.31 ± 3.09	81.32 ± 3.3	89.42 ± 1.95	89.78 ± 2.03
Vote	96.82 ± 2.47	96.95 ± 2.14	94.02 ± 2.78	98.27 ± 1.48	98.34 ± 1.36
Vowel	90.74 ± 2.37	95.58 ± 1.12	96.15 ± 1.04	99.59 ± 0.28	99.59 ± 0.28
Waveform-5000	81.29 ± 1.43	95.29 ± 0.68	95.14 ± 0.67	93.81 ± 0.82	94.81 ± 0.73
Zoo	88.43 ± 2.7	89.48 ± 2.37	88.68 ± 2.66	89.48 ± 2.37	89.48 ± 2.37
Mean	85.71 ± 4.32	89.50 ± 3.52	87.88 ± 4.47	88.99 ± 3.69	89.39 ± 3.62

Table 3. Results of two-tailed *t*-test on AUC.

	C4.4	NB	SBC	CL-TAN
NB	23/8/5			
SBC	14/17/5	5/22/9		
CL-TAN	19/13/4	9/20/7	12/21/3	
FAN	20/12/4	10/18/8	14/21/1	6/28/2

(iii) FAN achieves a considerable improvement over CL-TAN (six wins and two losses), but its advantage over naive Bayes is still not clear (ten wins and eight losses).
(iv) Naive Bayes achieves the highest average 89.50% over all data sets among all the algorithms.

The experimental results show that naive Bayes actually performs well in ranking, even better than it does in classification. What are the reasons behind this? In the next section, we will explore the reasons.

4. The optimality of naive Bayes in ranking

Although naive Bayes performs well in classification, its learnability is very limited. In the binary domain, it can learn only linearly separable functions (Duda and Hart 1973). Moreover, it cannot learn even all the linearly separable functions. For example, Domingos and Pazzani (1997) discovered that several specific linear functions, such as conjunctive concepts and *m*-of-*n* concepts, are not learnable by naive Bayes. In other words, naive Bayes is not optimal in learning those concepts. However, we find that naive Bayes is optimal in ranking in both conjunctive concepts and *m*-of-*n* concepts. Here the optimality in ranking is defined as follows.

Definition 1: A classifier is called locally optimal on example E in ranking

(i) if E is a positive example, there is no negative example ranked after E
(ii) if E is a negative example, there is no positive example ranked before E.

Definition 2: A classifier is called globally optimal in ranking, if it is locally optimal on all examples in the example space of the given problem.

When a classifier is globally optimal, its AUC is always 1.

4.1 *Conjunctive concepts*

A conjunctive concept is a conjunction of n literals L_i, where a literal is a Boolean attribute or its negation. It has been shown that naive Bayes, as a classifier, is optimal in learning conjunctive concepts if examples are uniformly distributed and the training set includes all 2^n possible examples (Domingos and Pazzani 1997). Let $+$ and $-$ denote the class of $C=1$ (true) and the class of $C=0$ (false), respectively. In the training set, only one example which has $L_1 = L_2 = \cdots = L_n = 1$ is in class $+$. Thus $p(+) = 1/2^n$, $p(-) = 2^n - 1/2^n$, $p(L_i|+) = 1$, $p(\bar{L}_i|+) = 0$, $p(\bar{L}_i|-) = 2^{n-1}/2^n - 1$, and $p(L_i|-) = (2^{n-1} - 1)/(2^n - 1)$. Assume that E is an arbitrary example and m is the number of its conjunction literals that are true. Then the class probability estimates given by naive Bayes are

$$p_{nb}(+|E) = p(+)p^m(L_i|+)p^{n-m}(\bar{L}_i|+)$$
$$= \begin{cases} \dfrac{1}{2^n} & \text{if } m = n \\ 0 & \text{otherwise} \end{cases}$$

$$p_{nb}(-|E) = p(-)p^m(L_i|-)p^{n-m}(\bar{L}_i|-)$$
$$= \frac{2^n - 1}{2^n}\left(\frac{2^{n-1} - 1}{2^n - 1}\right)^m \left(\frac{2^{n-1}}{2^n - 1}\right)^{n-m}.$$

It is easy to show that naive Bayes will give the correct classification for all examples. Let us consider the ranking produced by naive Bayes. For a positive example E_+, we have $m = n$. The probability $p_{nb}(+|E_+)$ is $1/2^n$. For any negative example E_-, $m < n$, and $p_{nb}(+|E_-) = 0 < 1/2^n = p_{nb}(+|E_+)$. Thus naive Bayes never ranks a positive example before a negative example in the class probability based ranking. Therefore naive Bayes is optimal for conjunctive concepts under uniform distribution.

If the assumption that examples are uniformly distributed is removed, naive Bayes gives the correct classification for all the examples in class $-$, given a sufficient training set. However, for a positive example ($m = n$), the result will depend on the class distribution. If $p(+) < 1/2^n$, it is possible that naive Bayes will fail to assign a correct class to a positive example. Thus naive Bayes is not optimal in classification if the example distribution is not uniform.

However, no matter what the value of $p(+)$, $p_{nb}(+|E_-) = 0$ and $p_{nb}(+|E_+) = p(+) > 0$. Therefore naive Bayes is still optimal for conjunctive concepts in ranking, as shown in the theorem below.

Theorem 1: *Naive Bayes is globally optimal in ranking on conjunctive concepts.*

4.2 *m-of-n concepts*

An m-of-n concept is a Boolean function that is true if m or more out of n Boolean attributes are true. Clearly, it is a linearly separable function. Domingos and Pazzani (1997) show that for the concept 8-of-25, when the input Boolean attributes have just six or seven 1s, naive Bayes gives an incorrect answer of 1 (instead of 0). They used the training set consisting of all 2^{25} examples of the 8-of-25 function. Let q denote $p(A_i = 1|+)$, where A_i is an attribute. Obviously, $q > 0.5$. The class probability estimate $p_{nb}(+|E)$ produced by naive Bayes is

$$p_{nb}(+|E) = p(+)q^i(1 - q)^{(n-i)}$$

where i is the number of attributes of 1. Assume that E_+ is a positive example with k_1 attributes of 1, and that E_- is a negative example with k_2 attributes of 1. Obviously, $k_1 \geq m > k_2$. Then we have

$$p_{nb}(+|E_+) - p_{nb}(+|E_-) = p(+)q^{k_2}(1 - q)^{n-k_1}(q^{k_1-k_2} - (1 - q)^{k_1-k_2}). \quad (8)$$

Since $q > 0.5$ and $k_1 > k_2$, equation (8) is always positive. Thus, for m-of-n concepts, the class probability of a positive example is always greater than the class probability of a negative example in naive Bayes. Therefore the ranking generated by naive Bayes is optimal, as shown in the following theorem.

Theorem 2: *Naive Bayes is globally optimal in ranking on m-of-n concepts.*

4.3 *General conditions for the optimality of naive Bayes*

The two preceding example problems are quite surprising, since it is known that, as a classifier, naive Bayes cannot learn all m-of-n concepts under uniform distribution and cannot learn all conjunctive concepts under some non-uniform distributions. However, the rankings generated by naive Bayes, are optimal in both problems. This provides evidence that naive Bayes performs well in ranking, and in some problems even better than classification.

In our following discussion, we assume that the prior probabilities $p(E)$ of all examples E are equal. Since

$$p(+|E) = \frac{p(+)p(E|+)}{p(E)},$$

the ranking is determined by $p(E|+)$.

Now let us consider the general case. Assume that E_+ is a positive example and E_- is a negative example. Thus $p(E_+|+) > p(E_-|+)$. Let $p_{nb}(E_i|+)$ denote the probability estimates generated by naive Bayes, $i = +, -$. Let x and y denote the errors of probability estimates on E_+ and E_-. i.e.

$$x = p(E_+|+) - p_{nb}(E_+|+)$$
$$y = p(E_-|+) - p_{nb}(E_-|+).$$

Naive Bayes generates the correct order for E_+ and E_- if

$$p_{nb}(E_+|+) > p_{nb}(E_-|+)$$

i.e.

$$y - x + (p(E_+|+) - p(E_-|+)) > 0. \tag{9}$$

Assuming that x and y are uniformly distributed, we plot a figure in which x and y correspond to the horizontal and vertical axes, respectively in figure 3. The shaded area corresponds to the cases in which equation (9) holds. Since $p(E_+|+) > p(E_-|+)$, naive Bayes is optimal in more than half of the possible area. It is easy to calculate the area A of the shaded area:

$$A = -\frac{1}{2}((p(E_+|+) - p(E_-|+)) - 2)^2 + 4. \tag{10}$$

It is interesting to note that the greater the difference between $p(E_+|+)$ and $p(E_-|+)$, the greater is the chance that naive Bayes is optimal. For example, when $p(E_+|+) - p(E_-|+) = 0.5$, the probability of naive Bayes being optimal is 0.78125.

Now let us assume that all the dependencies among attributes are complete. An attribute A_i is said to depend on A_j completely, if $A_i = A_j$. If $A_i = A_j$ and all other attributes are independent, the true probability $p(E|+)$ for an example $E = (a_1, a_2, \ldots, a_n)$ is

$$p(E|+) = p(a_i|+) \prod_{k \neq i,j} p(a_k|+).$$

The probability $p_{nb}(E|+)$ given by naive Bayes is

$$p_{nb}(E|+) = p(a_i|+)^2 \prod_{k \neq i,j} p(a_k|+).$$

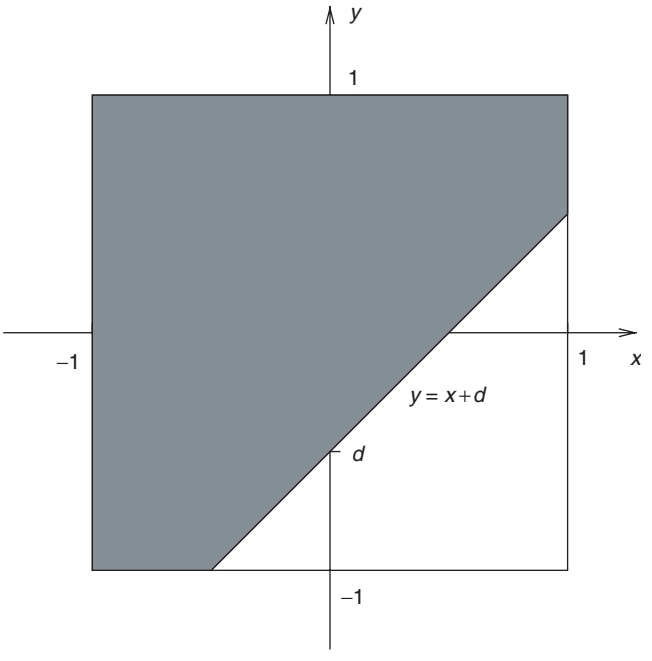

Figure 3. The optimality of naive Bayes in the general case in which $d = p(E_-|+) - p(E_+|+)$. The shaded area corresponds to the optimal area of naive Bayes.

Given two examples $E_+ = (a_1^+, a_2^+, \ldots, a_n^+)$ and $E_- = (a_1^-, a_2^-, \ldots, a_n^-)$ belonging to the positive and negative class respectively, we have

$$p(E_+|+) = p(a_i^+|+) \prod_{k \neq i,j} p(a_k^+|+) > p(E_-|+) = p(a_i^-|+) \prod_{k \neq i,j} p(a_k^-|+).$$

It is easy to show that, if $p(a_i^+|+) \geq 0.5$, $p_{nb}(E_+|+) > p_{nb}(E_-|+)$. Notice that E_+ is a positive example it is a reasonable assumption that $p(a_i^+|+) \geq 0.5$. We have a formal definition of the property of such an attribute value.

Definition 3: A value a_i of attributes A_i is called indicative to class c, if $p(A_i = a_i|c) \geq p(A_i = \bar{a}_i|c)$, where $\bar{a}_i$ is another value of A_i other than a_i.

For example, for the problem of m-of-n concepts, $p(A_i = 1|+) > p(A_i = 0|+)$ for any attribute. Therefore $A_i = 1$ is indicative to class $+$. If all the attribute values of an example are indicative, naive Bayes always gives the optimal ranking for it, illustrated by the theorem below.

Theorem 3: *Naive Bayes is optimal on example* $E = (a_1, a_2, \ldots, a_n)$ *in ranking, if each attribute value of E is indicative.*

By induction on i, the number of pairs of attributes with complete dependency.

When $i = 1$, it is true from the preceding discussion. Assume that the claim is true when $i = k$, i.e., if there are k complete dependencies among attributes and $p(E_+|+) > p(E_-|+)$, then $p_{nb}(E_+|+) > p_{nb}(E_-|+)$, where $E_+ = (a_1^+, a_2^+, \ldots, a_n^+)$ and $E_- = (a_1^-, a_2^-, \ldots, a_n^-)$, belong to the positive and negative class, respectively.

Consider $i = k + 1$. Assume that the new complete dependency is between A_{n-1} and A_n. Since $A_{n-1} = A_n$,

$$p(E_+|+) = p(E_+ - \{A_{n-1}\}|+) = p(a_1^+, \ldots, a_{n-2}^+, a_n^+|+)$$
$$p(E_-|+) = p(E_- - \{A_{n-1}\}|+) = p(a_1^-, \ldots, a_{n-2}^-, a_n^-|+).$$

Since there are only k dependencies among $A_1, \ldots, A_{n-2}, A_n$, according to induction hypothesis,

$$p_{nb}(a_1^+, \ldots, a_{n-2}^+, a_n^+| +) > p_{nb}(a_1^-, \ldots, a_{n-2}^-, a_n^-| +).$$

Thus we have

$$\prod_{i=1 i\neq n-1}^{n} p(a_i^+|+) > \prod_{i=1 i\neq n-1}^{n} p(a_i^-|+).$$

Since all the attribute values of E are indicative, $p(a_{n-1}^+|+) > p(a_{n-1}^-|+)$. Then, we have

$$\prod_{i=1}^{n} p(a_i^+|+) > \prod_{i=1}^{n} p(a_i^-|+).$$

Therefore, $p_{nb}(E_+|+) > p_{nb}(E_-|+)$.

Theorem 3 presents a sufficient condition on the local optimality of naive Bayes. Notice that even when all the attribute values of an example are indicative, it is possible that naive Bayes gives the wrong classification.

5. Conclusions

We have investigated the ranking performance of naive Bayes from both empirical and theoretical approaches. Our study suggests that naive Bayes performs well in ranking, even better than it does in classification. Our experiments show that naive Bayes is comparable with its extensions, such as SBC and TAN, and outperforms the state-of-the-art decision-tree learning algorithm C4.4 in terms of ranking. We investigated two example problems theoretically, conjunctive literals and m-of-n concepts, which were used to analyse the classification performance of naive Bayes by Domingos and Pazzani (1997). Surprisingly, naive Bayes works optimally in both problems with respect to ranking, although it does not perform optimally in classification. For more general cases, we proposed a sufficient condition for the local optimality of naive Bayes in ranking.

Ranking is a task between classification and probability estimation. It is similar to classification in the sense that both tolerate the estimation error of class probabilities to some extent; the performance in ranking of some learning algorithms, such as decision-trees, is quite different from that in classification. However, our study shows that naive Bayes performs more consistently. This is another attractive property of naive Bayes in the real world applications in which an accurate ranking is desired.

References

P.N. Bennett, "Assessing the calibration of naive Bayes' posterior estimates", Tech. Report No. CMU-CS00-155, School of Computer Science, Carnegie Mellen University, 2000.

A.P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms", *Pattern Recogn.*, 30, pp. 1145–1159, 1997.

P. Domingos and M. Pazzani, "Beyond independence: conditions for the optimality of the simple Bayesian classifier", *Mach. Learn.*, 29, pp. 103–130, 1997.

R.O. Duda and P.E. Hart, *Pattern Classification and Scene Analysis*, New York: Wiley-Interscience, 1973.

C. Ferri, P.A. Flach and J. Hernandez-Orallo, "Learning decision trees using the area under the ROC curve", in *Proceedings of the 19th International Conference on Machine Learning*, San Fransisco, CA: Morgan Kaufmann, 2002, pp. 139–146.

E. Frank, L. Trigg and G. Holmes, I.H. Witten, "Naive Bayes for regression", *Mach. Learn.*, 41, pp. 5–15, 2001.

J. Friedman, "On bias, variance, 0/1-loss, and the curse of dimensionality", *Data Min. Know. Discov.*, 1, pp. 55–77, 1995.

N. Friedman, D. Greiger and M. Goldszmidt, "Bayesian network classifiers", *Mach. Learn.*, 29, pp. 103–130, 1997.

D.J. Hand and R.J. Till, "A simple generalisation of the area under the ROC curve for multiple class classification problems", *Mach. Learn.*, 45, pp. 171–186, 2001.

L. Jiang, H. Zhang, Z. Cai and J. Su, "Learning tree augmented naive Bayes for ranking", in *Proceedings of the 10th International Conference on Database Systems for Advanced Applications*, Berlin Springer-Verlag, 2005, pp. 688–698.

I. Kononenko, "Comparison of inductive and naive Bayesian learning approaches to automatic knowledge acquisition", *Current Trends in Knowledge Acquisition*, Amsterdam: IOS Press, 1990.

P. Langley, W. Iba and K. Thomas, "An analysis of Bayesian classifiers", in *Proceedings of the Tenth National Conference of Artificial Intelligence*, Cambridge, MA: AAAI Press, pp. 223–228, 1992.

P. Langley and S. Sage, "Induction of selective Bayesian classifiers", in *Proceedings of Uncertainty in Artificial Intelligence 1994*, San Fransisco, CA: Morgan Kaufmann, 1994.

C.X. Ling, J. Huang and H. Zhang, "AUC: a statistically consistent and more discriminating measure than accuracy", in *Proceedings of the International Joint Conference on Artificial Intelligence IJCAI03*, San Farnsisco, CA: Morgan Kaufmann, 2003, pp. 329–341.

C.X. Ling and R. J. Yan, "Decision tree with better ranking", in *Proceedings of the 20th International Conference on Machine Learning*, San Fransisco, CA: Morgan Kaufmann, 2003, pp. 480–487.

C. Merz, P. Murphy and D. Aha, UCI repository of machine learning databases. Dept of ICS, University of California, Irvine, 1997, Available online at: http://www.ics.uci.edu/mlearn/MLRepository.html

A. Niculescu and R. Caruana, "Predicting good probabilities with supervised learning", in *Proceedings of the 22nd International Conference on Machine Learning*, Boston, MA: ACM Press, 2005, pp. 625–632.

M. Pazzani, P. Merz, C. Murphy, P. Ali, K. Hume and T. Brunk, "Reducing misclassification costs", in *Proceedings of the 11th International Conference on Machine Learning*, San Francisco, CA: Morgan Kaufmann, 1994, pp. 217–225.

J. Platt, "Probabilistic outputs for support vector machines and comparison to regularized likelihood methods", *Advances in Large Margin Classifiers*, A. Sela, P. Bartlett, B. Schölkopf and D. Schurmans, Eds., Cambridge, MA: MIT Press, 1999, pp. 61–74.

F. Provost and P. Domingos, "Tree induction for probability-based ranking", *Mach. Learn.*, 52, pp. 199–215, 2003.

F. Provost and T. Fawcett, "Analysis and visualization of classifier performance: comparison under imprecise class and cost distribution", in *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, Cambridge, MA: AAAI Press, pp. 43–48, 1997.

F. Provost, T. Fawcett and R. Kohavi, "The case against accuracy estimation for comparing induction algorithms", in *Proceedings of the Fifteenth International Conference on Machine Learning*, San Francisco, CA: Morgan Kaufmann, 1998, 445–453.

J.R. Quinlan, *C4.5: Programs for Machine Learning*. San Mateo: Morgan Kaufmann, CA, 1993.

J. Swets, "Measuring the accuracy of diagnostic systems", *Science*, 240, pp. 1285–1293, 1988.

I.H. Witten and E. Frank, *Data Mining*: *Practical Machine Learning Tools and Techniques with Java Implementation*, San Francisco, CA: Morgan Kaufmann, 2000.

B. Zadrozny and C. Elkan, "Obtaining calibrated probability estimates from decision trees and naive Bayesian classifiers", in *Proceedings of the 18th International Conference on Machine Learning*, San Francisco, CA: Morgan Kaufmann, 2001, pp. 609–616.

B. Zadrozny and C. Elkan, "Transforming classifier scores into accurate multiclass probability estimates", in *Proceedings of the Eighth International Conference on Knowledge Discovery and Data Mining*, Boston, MA: ACM Press, 2002, pp. 694–699.