Evolving Extended Naïve Bayes Classifiers[*]

Frank Klawonn
Department of Computer Science
University of Applied Sciences BS/WF
Salzdahlumer Str. 46/48
D-38302 Wolfenbuettel, Germany
f.klawonn@fh-wolfenbuettel.de

Plamen Angelov
Department of Communication Systems
InfoLab21
Lancaster University
South Drive, Lancaster LA1 4WA, UK
p.angelov@lancaster.ac.uk

Abstract

Naïve Bayes classifiers are a very simple, but often effective tool for classification problems, although they are based on independence assumptions that do not hold in most cases. Extended naïve Bayes classifiers also rely on independence assumptions, but break them down to artificial subclasses, in this way becoming more powerful than ordinary naïve Bayes classifiers. Since the involved computations for Bayes classifiers are basically generalised mean value calculations, they easily render themselves to incremental and online learning. However, for the extended naïve Bayes classifiers it is necessary, to choose and construct the subclasses, a problem whose answer is not obvious, especially in the case of online learning. In this paper we propose an evolving extended naïve Bayes classifier that can learn and evolve in an online manner.

1. Introduction

In recent years, it was realised in data analysis and mining that there is a strong need for analysing and mining streaming data. Streaming data differ from the classical paradigm of data analysis in the sense that the data set to be analysed is not completely available at the start of the analysis. Data arrive as a stream. In most applications it is not possible to wait until a large amount of data has been collected. Instead, the analysis should be started as soon as possible, even with a small data set. However, when more data arrive, the analysis should not be re-started completely, but should be carried out in an incremental fashion for which new or modified learning algorithms are needed. In most cases, it is also impossible to store the full history of the data stream. Therefore, an algorithm for streaming data should only rely on a small, information-compressed excerpt from the original data stream. When a data stream is analysed in a purely incremental fashion, it is assumed that the underlying model of the data and its parameters do not change over time, which turns out to be an unrealistic assumption in many applications. Therefore, a proper analysis of a data stream must be able to evolve, i.e. to adjust its model structure and the corresponding parameter set, when changes in the data stream occur.

In this paper we focus on evolving classifiers, ie. we want to predict a categorical attribute (the class) based on the values of a number of input attributes. The correct classification is not available at the time, when the input data arrive and the prediction is needed. In order to be able to train the evolving classifier, we assume that the correct classification will be available at a later stage. For instance, when we want to predict whether it will rain the next day, the prediction must be based on known measurements available before the next day. Nevertheless, after the next day, it is known whether the prediction was correct or not.

Many traditional classifiers try to minimize the classification error directly. This means in most cases that the full information from a training data set is needed to build such classifiers. In an evolving setting it is not realistic to assume that all historical data can be stored. Although there are approaches to evolving, classification performance driven classifiers, they tend to be quite complex. Naïve Bayes classifiers do not suffer from this problem, they are distribution driven. Distributions and their parameters are very easy to track and to be adapted in an evolving fashion. However, a standard naïve Bayes classifier is very limited and not well-suited for more difficult classification problems, especially since only one simple joint probability distribution is used per class. Extended naïve Bayes classifiers allow the introduction of artificial subclasses for each class in order to achieve a better performance. Another advantage of extended naïve Bayes classifiers is that they can be interpreted as a rule-based fuzzy classifier so that their classification behaviour is easier to understand for non-experts.

The evolving extended naïve Bayes classifier proposed in this paper can handle continuous as well as categorical attributes. Updating the parameters of the probability distribution can be carried out in a purely incremental fashion, but it is also possible to neglect the influence of classifications that were learned a long time ago. The more interesting evolving part of the classifier is the introduction of new artificial subclasses in order to improve its performance.

Section 2 briefly reviews the concepts of standard and extended naïve Bayes classifiers and their relation to fuzzy classifiers. How incremental learning and evolving strategies can be applied to extended naïve Bayes classifiers is discussed in section 3, followed by a brief example in section 4. We conclude the paper by outlining future work, especially ideas for simplifying extended naïve Bayes classifiers in an evolving fashion.

2. Simple & Extended Naïve Bayes Classifiers

Supervised classification refers to the problem to predict the value of a categorical attribute, the so-called class, based on the values of other attributes that can be categorical, ordinal or continuous. A typical setting for supervised classification is where a data set of classified data objects is available for the construction of the classifier. Usually the data set is split up into a training and a test set, even multiple times in the case of cross-validation, in order to better judge the prediction quality of the classifier for unknown data. Typically, the misclassification rate is used as an indicator for the quality of the classifier. However, this is only a special case of a general cost or loss matrix that specifies the estimated average costs that will result from misclassifications of an object from each true class A to any other class B. The misclassification rate simply uses a cost matrix with ones everywhere, except in the diagonal. The costs in the diagonal, i.e. the losses for correct classifications are assumed to be zero. The costs for misclassifications can differ extremely so that simply counting the misclassifications might mislead the classifier. If, for instance, we want to predict whether a component of a safety critical system like an aeroplane will work without failure, the costs for misclassifying the component as faulty, although it would work, are the costs for exchanging the component. The costs for misclassifying a faulty component as correct can cause the death of many people as "costs".

A large variety of classifiers exists in the literature. For instance, linear discriminant analysis is a very simple statistical classifier that is easy to construct. Decision trees are very popular because they are easy to understand according to their simple rule-like structure. The same applies to fuzzy classifiers. Both of them rely on more complex learning algorithms. Naïve Bayes classifiers, that will be the focus of this paper, are also very elementary probabilistic models.

2.1. Naïve Bayes Classifiers

A Bayes classifiers exploits the well-known Bayes rule:

$$P(H|E) = \frac{P(E|H) \cdot P(H)}{P(E)} \qquad (1)$$

H is the hypothesis, in the case of classification any value of the categorical attribute that we want to predict. E is the evidence, i.e. the information from the observed attributes that we want to exploit to predict the considered categorical attribute. When want to minimize the misclassification rate, then simply the class c yielding the highest posterior probability $P(H = c|E)$ is predicted. If a cost matrix K for misclassifications is known, the prediction tries to minimize the expected loss. If the categorical attribute to be predicted can take values from the finite set of classes $\mathcal{C}$, then the entry $K(c, c')$ (for $c, c' \in \mathcal{C}$) in the cost matrix stands for the resulting costs, when predicting c' instead of the correct class c. We always assume $K(c, c) = 0$ for any $c \in \mathcal{C}$. Correct predictions do not cause any loss. The expected loss given evidence E and predicting class c is

$$
\begin{aligned}
\text{loss}(c|E) &= \sum_{c' \in \mathcal{C}} P(c'|E) \cdot K(c', c) \\
&= \sum_{c' \in \mathcal{C}} \frac{P(E|c') \cdot P(c')}{P(E)} \cdot K(c', c). \quad (2)
\end{aligned}
$$

In this case, the class c yielding the lowest value in (2) is predicted by the Bayes classifier. Note that in both equations (1) and (2), the probability $P(E)$ of the specific evidence does not have any influence on the predicted class c, since $P(E)$ can be considered as a constant factor independent of c. Therefore, for the decision in (1), it is sufficient to consider the likelihoods (unnormalized probabilities)

$$L(H|E) = P(E|H) \cdot P(H) \qquad (3)$$

and in (4) we only need the relative expected losses

$$
\begin{aligned}
\text{rloss}(c|E) &= \sum_{c' \in \mathcal{C}} L(c'|E) \cdot K(c', c) \\
&= \sum_{c' \in \mathcal{C}} P(E|c') \cdot P(c') \cdot K(c', c). \quad (4)
\end{aligned}
$$

The evidence E represents the measured or given values of the attributes that we exploit for the prediction of the considered categorical attribute. For a naïve Bayes classifier (see for instance [12]) it is assumed that these attributes are independent given the classes. This means that, if $A_1, \ldots, A_k$ are the attributes used for prediction, we have

$$
\begin{aligned}
P(A_1 = a_1, \ldots, A_k = a_k|c) &= \\
P(A_1 = a_1|c) \cdot \ldots \cdot P(A_k = a_k|c) & \qquad (5)
\end{aligned}
$$

In order to apply a naïve Bayes classifier, the probabilities in (5) must be known. In case that the attributes $A_1, \ldots, A_k$ are categorical attributes, these probabilities can be estimated based on the corresponding relative frequencies in the available training data set. For a continuous attribute A_i, it is necessary to estimate its (conditional) probability density function (pdf) $f_{A_i|c}$ from the training data set. Once we have an estimation for the pdf $f_{A_i|c}$, the corresponding value $f_{A_i|c}(a_i)$ is used in (5) instead of the probability $P(A_i = a_i|c)$. This means the computed value $P(A_1 = a_1, \ldots, A_k = a_k|c)$ is no longer a probability, but a likelihood, i.e. an unnormalized probability. In order to estimate the pdf $f_{A_i|c}$, it is usually assumed that $f_{A_i|c}$ belongs to a class of parametric distributions, so that only the corresponding parameters of the pdf have to be estimated. A very typical assumption is that $A_i|c$ is normally distributed with unknown mean $\mu_{i|c}$ and unknown variance $\sigma^2_{i|c}$. Such parameters can be estimated from the training data set using the corresponding well-known formulae from statistics.

2.2. Extended Naïve Bayes Classifiers

The underlying model of a naïve Bayes classifier assumes that each class is characterised by a probability distribution and for each class this probability distribution is simply the product of it marginals. Figure 1 shows the normal distributions for the well known iris data set [3] learned by a naïve Bayes classifier. The categorical attribute to be predicted has three different values for the iris data set. Four continuous attributes are used for the prediction.
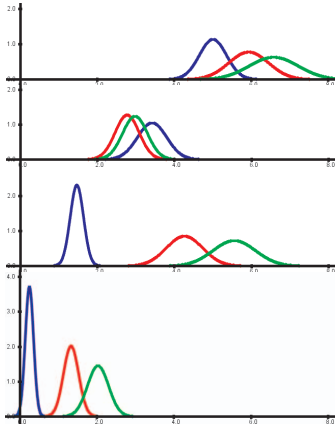

Figure 1. The normal distributions for the three classes for each of the four attributes for the iris data set.

The classification performance will work out well in cases like the iris data set, when the distribution of the data can be described roughly by a (multinomial) normal distri-

bution for each class. If, however, the data objects of one class do not form a compact cluster, but are distributed over different separated clusters, then a naïve Bayes classifier might fail completely. In this sense a naïve Bayes classifier can be viewed as a specific form of a fuzzy classifier using exactly one rule per class [7]. For this purpose the probability distributions must be scaled in such a way that they never yield values greater than one, in order to interpret them as fuzzy sets. This leads to a constant scaling factor that does not influence the classification decision. The fuzzy sets for one class are aggregated by the product operator. Typically, fuzzy classifiers use the minimum for this operation. However, the minimum often leads to severe restrictions [11] and the product is a possible alternative. The prior probabilities for the classes correspond to rule weights.

In order to overcome this restricted assumption of unimodal distributions of the data for each class, the introduction of pseudo-classes was proposed in [6]. In the sense of a fuzzy classifier, more than one rule is allowed for each class. In terms of a Bayes classifier, each class is represented by a number of artificial subclasses. In order to classify a data object, such an extended naïve Bayes classifier will first compute the posterior probabilities (likelihoods) for each pseudo-class in the same way as a standard naïve Bayes classifier. The posterior probabilities (likelihoods) for the actual classes are simply obtained as the sums of the posterior probabilities (likelihoods) of the corresponding pseudo-classes.

Although the classification of a new data object is obvious for an extended naïve Bayes classifier, it is not clear at all, how to estimate the prior probabilities and the corresponding probability distributions for the pseudo-classes. The problem during the training phase or construction of the extended naïve Bayes classifier is to decide to which of its subclasses an object of a specific class should be assigned. The problem will be treated in the following section.

3. Incremental Learning and Evolving Extended Naïve Bayes Classifiers

Probabilistic models are almost perfectly suited foronline learning, since most statistical concepts rely on counting principle that render easily to online implementations.

3.1. Advantages of Weighted Mean Concepts

A mean value $[x]_{(n)} = \frac{1}{n} \sum_{i=1}^n x_i$ can be updated easily in an incremental manner by

$$[x]_{n+1} = \frac{n}{n+1}[x]_n + \frac{1}{n+1}x_{n+1}. \qquad (6)$$

To update a mean value, it is sufficient to know the mean value of the previous observations, the new observation and

the number of observations. Note that the concept of a mean value, respectively its estimation, is much more general. It applies also to derived concepts, especially to statistical moments. This fact can be exploited to compute variances and covariances in an online fashion. For the empirical variance, we have

$$\hat{\sigma}_n^2 = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - [x]_n)^2 = \frac{n}{n-1} \left([x^2]_n - [x]_n^2 \right) . \quad (7)$$

This means, the variance only involves the computation of the two means $[x^2]_n$ and $[x]_n$, i.e. the first two moments.

The general idea of the Bayesian approach in statistics is to update probabilities based on new observations. Therefore, it is no wonder that there exist many incremental learning approaches in the spirit of the general Bayesian idea [1, 5]. These concepts are applied to many statistical methods like linear discriminant analysis [8], but also to Bayesian networks [4], classifiers [10] and especially to naïve Bayes classifiers [9].

So far, we have only focused on incremental learning. Evolving systems do not only learn in an online fashion, but also adapt and change their structure while new data arrive. It is also important to notice changes in the data stream. A simple incremental strategy as (6) will slow down the learning process with an increasing number of data.

(6) is nothing else than a specific convex combination of the previous mean $[x]_n$ and the new observation x_{n+1}. One can also think of other convex combinations, for instance $[x]_{n+1} = (1 - \alpha) \cdot [x]_n + \alpha \cdot x_{n+1}$ with a fixed constant $\alpha \in [0, 1]$. This corresponds to the well known idea of exponential decay of the information. After k new observations the contribution of an observation to the mean will decrease in an exponential fashion to $(1 - \alpha)^k \cdot \alpha$. One can also realise a sliding window concept of size k for the mean easily, if it is possible to store a fixed number of k data objects for the classifier. In the sense of the Bayesian approach this means that new observations are expected to follow the same distribution as the previous k ones.

3.2. Incremental Learning

The above mentioned concepts can be applied to a naïve Bayes classifier in a straight forward manner. The prior probabilities $P(c)$ for the classes are nothing else than mean values for counting variables, so that (6) can be used. The same applies to the conditional probabilities $P(A_i|c)$ for discrete attributes A_i. For continuous attributes it is necessary to assume a certain type of distribution. When the distribution is determined by a finite number of its moments, like a normal distribution by its first two moments, then again (6) is applicable. It is not necessary to consider complicated distributions like instance Gaussian mixtures, since mixtures can be modelled by the pseudo-classes.

So far, it is not clear which class to choose for updating the corresponding probability distributions. For a standard naïve Bayes classifier, the probability distributions for the class of the new observed object is updated. But since in an extended naïve Bayes classifier one class might be represented by a number of pseudo-classes, we have to choose one of the pseudo-classes. In case, the extended naïve Bayes classifier has predicted the correct class for the new observed object, we update the probability distributions of the corresponding pseudo-class that had yielded the highest likelihood. We could also choose this pseudo-class in case of a misclassification by choosing the pseudo-class with the highest likelihood from all pseudo-classes representing the correct class. However, we slightly deviate from this concept in case of misclassification. The reason is the way we construct the pseudo-classes in an online fashion.

3.3. Evolving Extended Naïve Bayes Classifiers

The evolving extended naïve Bayes classifier is initialised one pseudo-class per class. New pseudo-classes are introduced, when the misclassification rate or the average loss is too high. Since the misclassification rate and the average loss per class are also weighted mean concepts, they can be tracked in an online fashion.

When the misclassification rate or the average loss is too high and we decide to introduce a new pseudo-class, we assign the pseudo-class to the class for which we have the highest misclassification rate or the highest average loss, respectively. We also have to specify initial probability distributions for the attributes for the new pseudo-class and also the prior probability for the new pseudo-class. For the introduction of new pseudo-classes, we apply the following strategy. When we create the initial naïve Bayes classifier, we already introduce for each class a second hidden pseudo-class. These pseudo-classes are not used for prediction. But we update the probability distributions for a hidden pseudo-class each time a misclassification occurred for an object of the associated class. In this way, the hidden pseudo-classes can already start to adapt their probability distributions to those objects that are misclassified, although they do not participate in the classification process. When a hidden pseudo-class is added to the actual pseudo-classes, we simply use the probability distributions for the attributes that were computed for the hidden class. The prior probability for the new pseudo-class is calculated as follows. Note that we cannot simply use the tracked prior probability of the hidden pseudo-class, since otherwise the sum of the prior probabilities of all classes would exceed one. We simply interpret the prior probabilities of the old pseudo-classes and the tracked prior probability of the new class as likelihoods and carry out a normalisation. This will give at least a small chance to the added pseudo-class to yield the high-

est likelihood. When we move a hidden pseudo-class to the extended naïve Bayes classifier, we introduce a new hidden pseudo-class associated with the same class to the hidden pseudo-classes. The prior probabilities of this new hidden pseudo-class are initialised with standard parameters. The prior probability is set to zero.

When a new object is classified incorrectly by the extended naïve Bayes classifier, we update the probability distributions of that pseudo-class associated with the correct class that was last added to the extended naïve Bayes classifier. When we have moved a hidden pseudo-class to the extended naïve Bayes classifier, we do not immediately move another hidden pseudo-class to the extended naïve Bayes classifier after the next observation, since in most cases the misclassification rate or average loss will not drop immediately. Before we move another hidden pseudo-class to the extended naïve Bayes classifier, we wait a fixed number of new observations in order to give the modified classifier a chance to adapt its parameters.

4. An Application Example

In order to illustrate how our approach works, we consider a modified version of the iris data set. The original iris data set contains three classes that are roughly grouped into three clusters. We artificially join two of the classes, so that one class is represented by two clusters. We then apply our incremental evolving extended naïve Bayes classifier to a stream of randomly drawn samples from the iris data set. The classifier constructs four pseudo-classes, two for each class, i.e. one more than expected. Figure 2 shows the corresponding normal distributions. Although the distributions do not completely look like one would expect, the misclassification rate is quite low (4%). The left normal distribution for the first attribute and the corresponding pseudo-class is almost not used by the classifier.
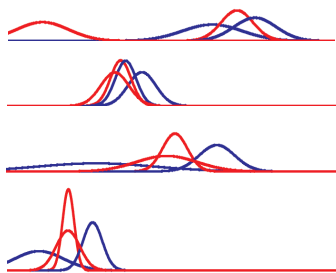

Figure 2. The normal distributions for the four pseudo-classes resulting from the modified iris data set.

5. Conclusions

In this paper, we have proposed an extended version of a naïve Bayes classifier that is able to learn in an incremental fashion and to extend its structure automatically, when the data from the data stream cannot be classified well enough. Future work will include concepts to reduce the number of pseudo-classes. Pseudo-classes with an extremely low prior probability can be removed. Also pseudo-classes with similar probability distributions can be joint together. Here a χ^2-test could be applied. We also plan to incorporate ideas from semi-supervised online learning [2], in case the classification is not available for all data.

References

[1] J. Anderson and M. Matessa. Explorations of an incremental, Bayesian algorithm for categorization. *Machine Learning*, 9:275–308, 1992.

[2] P. Angelov and D. Filev. Flexible modells with evolving structure. *International Journal on Intelligent Systems*, 19:327–340, 2004.

[3] C. Blake and C. Merz. UCI repository of machine learning databases. http://www.ics.uci.edu/~mlearn/MLRepository.html, 1998.

[4] N. Friedman and M. Goldszmidt. Sequential update of Bayesian network structure. In *Proc. 13th Conference on Uncertainty in Artificial Intelligence*, pages 165–174, San Francisco, 1997. Morgan Kaufmann.

[5] J. Gama. Iterative Bayes. *Theoretical Computer Science*, 292:417–430, 2003.

[6] A. Klose. *Partially Supervised Learning of Fuzzy Classification Rules*. Ph.D. thesis, Otto-von-Guericke-Universität Magdeburg, 2004.

[7] A. Nürnberger, C. Borgelt, and A. Klose. Improving naive Bayes classifiers using neuro-fuzzy learning. In *Proc. 6th International Conference on Neural Information Processing (ICONIP99)*, page 154159, Perth, 1999.

[8] S. Pang, S. Ozawa, and N. Kasabo. Incremental linear discriminant analysis for classification of data streams. *IEEE Transactions on Systems, Man, and Cybernetics – Part B*, 35:905–914, 2005.

[9] J. Roure. Incremental learning of tree augmented naive Bayes classifiers. In F. Garijo, J. Riquelme, and M. Toro, editors, *Proc. 8th Ibero-American Conference of Artificial Intelligence: IBERAMIA 2002*, pages 32–41, Berlin, 2002. Springer.

[10] M. Singh and M. Gregory. Efficient learning of selective Bayesian network classifiers. In L. Saitta, editor, *Proc. 13th International Conference on Machine Learning*, pages 453–461, San Francisco, 1996. Morgan Kaufmann.

[11] B. von Schmidt and F. Klawonn. Fuzzy max-min classifiers decide locally on the basis of two attributes. *Mathware and Soft Computing*, 6:91–108, 1999.

[12] I. Witten and E. Frank. *Data Mining*. Morgan Kaufmann, San Francisco, 2nd edition, 2005.