Hierarchical Naive Bayes Classifiers for uncertain data*

Riccardo Bellazzi[1], Francesca Demichelis[2], Paolo Piergiorgi[1,3], and Paolo Magni[1]

[1] Dipartimento di Informatica e Sistemistica, Universita' di Pavia, via Ferrata 1, 27100 Pavia, Italy
{riccardo.bellazzi, paolo.magni}@unipv.it
http://www.labmedinfo.org
[2] Brigham and Women's Hospital Department of Pathology, LMRC 414 221 Longwood Avenue Boston, MA 02115, USA
fdemichelis@partners.org
[3] Daisy-Labs, Piacenza, Italy
piergiorgi@gmail.com

Abstract. In experimental sciences many classification problems deal with variables with replicated measurements. In this case the replicates are usually summarized by their mean or median. However, such choice does not consider the information about the uncertainty associated with the measurements, thus potentially leading to over or underestimate the probability associated to each classification. In this paper we present an extension of the Naive Bayes classifier which, thanks to a Bayesian hierarchical model, is able to properly deal with replicates and uncertain measurements. We will show how to perform classification and learning with continuous and discrete variables with replicated measurements and we will describe the advantages of the proposed model over the standard Naive Bayes algorithm with a simulation study.

1 Introduction

One of the fundamental topics of Supervised Machine Learning is the automated construction of classifiers from labelled data, which can be then used to forecast the class of a new example, given the values of its attributes. The exploitation of a classification model may be very useful when dealing with experimental data, such as in biomedical applications, where the classifier can eventually be used for diagnostic and prognostic purposes. In many experimental contexts, however, the data belonging to a single example are the results of the averaging of different repeated measurements, aimed at mitigating the effect of intra-example variability and the measurement error. This happens, for example, when dealing with biological samples, in which the datum of the variables of interest is obtained by repeating two or more times the same measurement procedure and

* This work is part of the FIRB project "Learning theory and engineering applications", funded by the Italian Ministry of Education

then by averaging the results. Although each value is usually provided with its standard error, there have been very few attempts to deal with such kind of "uncertain" data when building a classification model [1, 2], and none of them is routinely applied. Nevertheless, the knowledge about the spread of the replicated measurements may be crucial in both the learning and the classification phase. For example, when monitoring diabetic patients, an average blood glucose value of 100 mg/dl obtained by three measurements of 100, 50 and 150 is clinically different from the same average value coming from the measurements 100, 110, and 90 mg/dl. In this paper we will propose an extension of the Naive Bayes classifier [3, 4] which is able to deal with such kind of repeated and uncertain measurements. The proposed classifier will handle repeated measurements resorting to a Bayesian hierarchical model [5–7]. In this paper we will show how to build a Bayesian hierarchical classifier in both the cases of discrete and continuous gaussian variables. While the classification phase will exploit a close form equation for computing the posterior probability distribution, the learning phase will be implemented resorting to convenient approximations or to the EM algorithm. We will discuss the results obtained on a set of simulated data in comparison with the standard Naive Bayes approach.

2 The hierarchical Naive Bayes classifier

The hierarchical Naive Bayes classifier that we introduce in this section assumes that the measurements are stochastic variables with a hierarchical structure in terms of their probability distributions. We suppose that we can collect a number n_{rep} of observations, or *replicates* on each example, and that an example belongs to one of a set of given classes. Let us suppose that x is a stochastic variable representing the replicates, whose probability distribution is dependent on a vector of parameters θ, which corresponds to the single example, and may represent, for example, the mean and variance of the probability distribution of replicates; if we consider the i-th example, with i in $1, \ldots, N$, the probability distribution of the vector of the replicates is given by $p_{(X_i|\theta_i)}$, with $X_i = \{x_{i1}, ...x_{ij}, ..., x_{in_{rep}}\}$, while the probability distribution of the individual parameters is $p_{(\theta_i|\xi_{C_k})}$, where ξ_{C_k} is a set of population hyper-parameters that depends on the class C_k in the set $C = \{C_1, ...C_h\}$ to which the example belongs, and is thus the same for all the examples of the same class. Figure 1 shows the representation of the problems though a graphical model with plates [8].

In a Bayesian framework, the classification step is therefore performed by finding the class with the highest posterior probability distribution. Thanks to the conditional independence assumptions of the hierarchical model described above, we can write $P_{(C_k|X)} \propto P_{(X|\xi_{C_k})} P_{(\xi_{C_k}|C_k)} P_{(C_k)}$. Since the population parameters ξ_{C_k} are determined by the knowledge of the class C_k with probability one, the equation can be simplified as $P_{(C_k|X)} \propto P_{(X|\xi_{C_k})} P_{(C_k)}$

The posterior is thus dependent on the so-called *marginal likelihood*, $P_{(X|\xi_{C_k})}$, which can be calculated by integrating out the vector of parameters θ as follows:

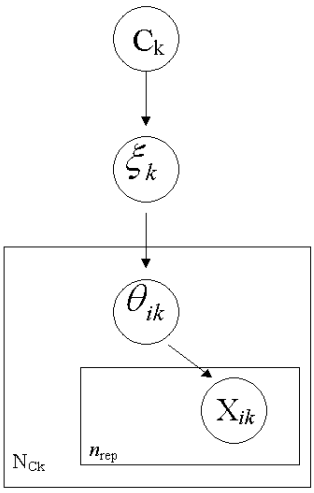

Fig. 1. The hierarchical structure of the data represented with the plates notation. Many replicates are available for each example. The examples are characterized by an individual vector of parameters θ and the examples belonging to the same class have a common set of parameters ξ.

$$P_{(X|\xi_{C_k})} = \int_{\Omega_\theta} P_{(X|\theta)} P_{(\theta|\xi_{C_k})} d\theta \tag{1}$$

where Ω_θ is the support of θ.

The learning problem will therefore consist in estimating the population parameters ξ_{C_k} for each class, while the classification problem is mainly related to the calculation of the marginal likelihood.

To deal with multivariate problems, in this paper we resort to the Naive Bayes algorithm, which assume that each attribute is conditionally independent from the others given the class.

$$P_{(C_k|X)} \propto P_{(C_k)} \prod_{f=1}^{N_{feature}} P_{(X_f|C_k)} \tag{2}$$

From the computational viewpoint, this will allow us to compute separately the marginal likelihood for each variable to perform classification and to learn a collection of independent univariate models.

In the following of the paper we will show how to deal with the classification and learning problem when i) the variables are continuous and normally distributed; ii) the variables are discrete with multinomial distribution. These two cases allow to deal with the majority of classification problems. Let us note

that we will perform our analysis considering a single attribute x; since we are exploiting a Naive Bayes strategy, the results will be easily generalized to the multivariate case thanks to (2).

3 Continuous variables with Gaussian distribution

Let us denote x_{ij} the observed value of the j-th replicate of the i-th example, with $j = 1, 2, ..., n_{rep_i}$ and $i = 1, 2, ..N_{C_k}$ where N_{C_k} is the number of examples belonging to the class C_k.
We assume that the replicates of the i-th examples are Gaussian distributed stochastic variables with mean μ_i and variance σ^2; we also assume that the variance depends only on the class C_k. Following the hierarchical model described in the previous section, the average values of the different examples belonging to the same class are normally distributed with mean M and variance τ^2.

$$x_{ij} \sim N_{(\mu_i,\sigma^2)} \tag{3}$$

$$\mu_i \sim N_{(M,\tau^2)} \tag{4}$$

The hierarchical model corresponds to the general model described in the previous section (see Figure 1) with $\theta = \{\mu\}$ and $\xi = \{M, \tau^2, \sigma^2\}$.

3.1 Classification

As described in Section 2, the classification problem needs the computation of the marginal likelihood $P_{(X|\xi_{C_k})}$, where X is the example to be classified, for which $x_1, \ldots, x_{n_{rep}}$ replicates are available. In this case, given the conditional independence model of Figure 1, we can write

$$P_{(X|\xi_{C_k})} = P_{(X|\sigma^2,M,\tau^2)} = \int_{\Omega_\mu} P_{(X|\mu,\sigma^2)} P_{(\mu|M,\tau^2)} d\mu \tag{5}$$

where we integrate over μ only, since we have assumed that σ^2 is constant over the class.

The integral of equation (5) can be solved as follows:

$$P_{(X|\sigma^2,M,\tau^2)} = \frac{\sigma}{(\sqrt{2\pi}\sigma)^{n_{rep}}(\sqrt{n\tau^2+\sigma^2})} e^{-\frac{1}{2}(\frac{M^2}{\tau^2}+\frac{\sum_{j=1}^{n_{rep}} x_j^2}{\sigma^2})}$$
$$* e^{\frac{1}{2}\frac{\frac{\sigma^2 M^2}{\tau^2}+\frac{n_{rep}^2 \bar{x}^2 \tau^2}{\sigma^2}+2\bar{x}M n_{rep}}{n_{rep}\tau^2+\sigma^2}} \tag{6}$$

where $\bar{x} = \frac{\sum_{j=1}^{n_{rep}} x_j}{n_{rep}}$. Given the marginal likelihood of each feature, we can compute the posterior probability distribution as in equation (2).

The computation of the marginal likelihood requires the knowledge of the population (class) parameters $\xi = (\sigma, \tau, M)$, which can be learned from the data resorting to different strategies.

3.2 Empirical learning

A fast strategy for calculating an estimate of the model parameters (σ, τ, M), can be obtained with an approximation of the maximum likelihood estimate *(ML)* called *empirical learning* [9, 10].

We first estimate the parameters of each example through the calculation of the sample mean and variance:

$$\hat{\mu}_i = \frac{\sum_{j=1}^{n_{rep_i}} x_{ij}}{n_{rep_i}} \tag{7}$$

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^{N_{C_k}} \frac{\sum_{j=1}^{n_{rep_i}} (x_{ij} - \hat{\mu}_i)^2}{n_{rep_i}}}{N_{C_k}} = \frac{\sum_{i=1}^{N_{C_k}} \hat{\sigma}_i^2}{N_{C_k}} \tag{8}$$

where $\hat{\sigma}_i = \frac{\sum_{j=1}^{n_{rep_i}} (x_{ij} - \mu_i)^2}{n_{rep_i}}$ is the variance of the i-th example.

The population parameters are then obtained as:

$$\hat{M} = \frac{\sum_{i=1}^{N_{C_k}} \hat{\mu}_i n_{rep_i}}{\sum_{i=1}^{N_{C_k}} n_{rep_i}} \tag{9}$$

$$\hat{\tau}^2 = \frac{\sum_{i=1}^{N_{C_k}} (\hat{\mu}_i - \hat{M})^2}{N_{C_K}} - \frac{\sum_{i=1}^{N_{C_k}} \frac{\hat{\sigma}_i^2}{n_{rep_i}}}{N_{C_k}} \tag{10}$$

In this case the population mean is computed as a weighted average of the individual means. The weight increases when the number of available replicates increases. The population variance is derived by the subtraction of an estimate of the intra-example variance from the estimate of the inter-example variance. Of course, such estimate is valid only if the first term is greater than the second one, i.e. when the inter-example variance is greater than the intra-example one.

3.3 EM learning

A learning technique which better preserve the stochastic nature of the hierarchical model described in this section relies on the Expectation-Maximization (EM) strategy. Within this iterative strategy we will consider μ as a latent variable and $\xi = (M, \tau, \sigma)$ the non-latent ones.

EM starts with an initial guess on the non latent variables ξ^0. Then, an expectation and a maximization steps are iterated as follows.

E-step In the expectation step the expected values of the latent parameter is calculated as:

$$E[\mu_i|X,\xi^{t-1}] = \hat{\mu}_i = \frac{\frac{\hat{M}^{t-1}}{(\hat{\tau}^{t-1})^2} + \sum_{j=1}^{n_{rep_i}} \frac{x_{ij}}{(\hat{\sigma}^{t-1})^2}}{\frac{1}{(\hat{\tau}^{t-1})^2} + \frac{n_{rep_i}}{(\hat{\sigma}^{t-1})^2}} \tag{11}$$

$$Var[\mu_i|X,\xi^{t-1}] = Var[\mu_i] = \frac{1}{\frac{1}{(\hat{\tau}^{t-1})^2} + \frac{n_{rep_i}}{(\hat{\sigma}^{t-1})^2}} \tag{12}$$

where t is the index of the iteration.

M-step Once the expected value of μ is calculated, the maximization step finds the maximum likelihood estimate for the non-latent parameters. In this case the estimate is:

$$\hat{M}^t = \frac{\sum_{i=1}^{N_{C_k}} \hat{\mu}_i}{N_{C_k}} \tag{13}$$

$$(\hat{\tau}^t)^2 = \frac{\sum_{i=1}^{N_{C_k}} (\hat{\mu}_i - \hat{M}^t)^2 + Var[\mu_i]}{N_{C_k}} \tag{14}$$

$$(\hat{\sigma}^t)^2 = \frac{\sum_{i=1}^{N_{C_k}} \sum_{j=1}^{n_{rep_i}} (x_{ij} - \hat{\mu}_i)^2 + Var[\mu_i]}{\sum_{i=1}^{N_{C_k}} n_{rep_i}} \tag{15}$$

The iteration of the two steps guarantees the convergence of the parameter estimate to the maximum likelihood one.

4 Discrete variables

Although Gaussian distributed variables are rather common in nature, in particular after normalization and/or log-transformation, classification problems must often deal with qualitative variables or non-gaussian data that can be conveniently discretized. As a matter of fact, the Naive Bayes classifier is usually applied with discrete or discretized variables. For this reason, we herein propose a version of the Hierarchical Naive Bayes classifier for discrete variables. For sake of readability we have omitted the dependence of the vectors to the class k.

We assume that the vector of the occurrences (counts) of the i-th example is $X_i = \{x_{i1}, ..., x_{ij}, ..., x_{iS}\}$, where x_{ij} is the number of occurrences of the j-th discrete value, or state, of the i-th example and S is the number of states of the variable x. The number of replicates of each example is given by $n_{rep_i} = \sum_j^S x_{ij}$. We also assume that the relationship between the data X_i and the example parameters θ_i is expressed by a multinomial distribution:

$$X_i \sim Multin(n_{rep_i}, \theta_{i1}, ..., \theta_{ij}, ..., \theta_{iS}) \tag{16}$$

Therefore θ_i is an S-dimensional vector, where θ_{ij} represents the probability of the occurrence of the j-th event in the example i. The parameters θ_i, for $i = 1, 2, ..., N_{C_k}$, are characterized by the same prior Dirichlet distribution:

$$\theta_i \sim Dirichlet(\alpha\xi_1, ..., \alpha\xi_S) \tag{17}$$

with probability density:

$$P(\theta_i|\alpha, \xi) = \frac{\Gamma_{(\alpha)}}{\prod_{j=1}^{S} \Gamma_{(\alpha\xi_j)}} \prod_{j=1}^{S} \theta_{ij}^{\alpha\xi_j - 1} \tag{18}$$

where $0 < \alpha < \infty$, $\xi_j < 1 \; \forall \; j = 1, ..., S$ and $\sum_{j=1}^{S} \xi_j = 1$. Following the hierarchical model of Section 2, the individual example parameters θ_i, are independent from each other given $\xi = \{\xi_1, ..., \xi_S\}$ and α.

In the following we will assume that the parameter α will be fixed, and it will be thus treated as a design parameters of the algorithm. α represents the prior assumption on the degree of similarity of all examples belonging to the same class. A proper setting of the parameter α allows to derive a compromise between a *pooling* strategy, where all replicates are assumed to belong to the same example and a full hierarchical strategy where all examples are assumed to be different.

4.1 Classification

As described in Section 2, the classification problem requires the computation of the marginal likelihood (1). We assume that an estimate of the population parameters ξ is available and that α is known. Given an example with counts distributed on different states $X = \{x_1, ..., x_S\}$, where $n_{rep} = \sum_{j=1}^{S} x_j$, we must compute:

$$P_{(X|C_k,\xi)} = \int_{\Omega_\theta} P_{(X|\theta)} P_{(\theta|\xi)} d\theta \tag{19}$$

where $\theta = \{\theta_1, ..., \theta_S\}$ is the vector of the individual example parameters, with $\sum_{j=1}^{S} \theta_j = 1$ and Ω_θ the support of θ.

This integral can be solved by noting that it is an integral of the product of a Multinomial and a Dirichlet distribution. The marginal likelihood can be thus computed as:

$$P_{(X|C_k,\xi)} = \frac{n_{rep}! \; \Gamma(\sum_i \alpha\xi_i)}{\Gamma(\sum_i x_i + \alpha\xi_i)} \prod_i \frac{\Gamma(x_i + \alpha\xi_i)}{x_i! \; \Gamma(\alpha\xi_i)} \tag{20}$$

The Naive Bayes approach allows to exploit this equation for each variable in the problem at hand, and then to apply the equation (2) to perform the classification. The marginal likelihood however requires the estimate of the population parameters ξ from the data. In analogy with Section 3, we will propose two different strategies to learn the model from a data set of examples.

4.2 Learning with collapsing

The task of learning the population parameters can be performed by resorting to approximated techniques. Herein we will describe a strategy previously presented by [11] and [12].

We suppose that a data set $X = \{X_1, ..., X_{N_{C_k}}\}$ is available for each class. Such vector is transformed into a new vector X^* where the i-th element $X_i^* = \{\tau_i x_{i1}, ..., \tau_i x_{ij}, ..., \tau_i x_{in_{rep_i}}\}$ with

$$\tau_i = \frac{1+\alpha}{n_{rep_i} + \alpha} \tag{21}$$

τ_i is a suitable weight that allows to take into account the prior assumptions on the heterogeneity of the example belonging to the class. The hierarchical model is then collapsed into a new model, where the vector of the measurements X_i^* is assumed to have a multinomial distribution with parameters ξ and $\tau_i n_{rep_i}$. Such assumption can be justified by the calculation of the first and second moment of $P_{(X^*|\xi)}$, which is computed by approximating the distribution of the parameters θ given ξ with its average value [11].

The ML estimate of the parameters ξ can be thus obtained for each state of the discrete variable as:

$$\hat{\xi}_j = \frac{\sum_{i=1}^{N_{C_k}} \tau_i x_{ij}}{\sum_{i=1}^{N_{C_k}} \tau_i n_{rep_i}} \tag{22}$$

Within this framework we can also provide a Bayesian estimate of the population parameters ξ. We assume that ξ is a stochastic vector with a Dirichlet prior distribution: $\xi \sim Dirichlet(\beta\gamma_1, ..., \beta\gamma_S)$, where $0 < \beta < \infty$, $\gamma_j < 1 \ \forall \ j = 1, ..., S$ and $\sum_{j=1}^{S} \gamma_j = 1$.

After collapsing, we may derive the posterior distribution of ξ is still a Dirichlet with expected value of the probability of the j-th state of the discrete variable:

$$\hat{\xi}_j = \frac{\sum_{i=1}^{N_{C_k}} \tau_i x_{ij} + \beta\gamma_j}{\sum_{i=1}^{N_{C_k}} \tau_i n_{rep_i} + \beta} \tag{23}$$

In this setting, the parameter vector γ and β assume the same meaning of the parameters usually specified in the Bayesian learning strategies applied in many Machine Learning algorithms. In particular, if we assume $\gamma = 1/S$ and $\beta = 1$ we obtain an estimate which is close to the Laplace estimate, while different choices of γ and β lead to estimates which are similar to the m-estimate, where β plays the role of m.

4.3 Learning with the EM strategy

As an alternative to the collapsing algorithm previously presented, we have applied an implementation of the EM strategy. In this case, however, the maximization step cannot be solved in closed form, so that we must run an iterative maximization algorithm within each M iteration, as proposed by [13].

In this case the latent variables are the parameters θ, and the non-latent ones are represented by the parameter vector ξ. As described in the previous chapter, EM starts with an initial guess on the non latent variables ξ^0. Then, an expectation and a maximization steps are iteratively performed.

E-step In the E-step we compute the expected value of each θ_i from the posterior distribution $P_{(\theta_i|X^*,\xi^{t-1})}$:

$$\theta_i \sim Dirichlet(\tau_1 x_{i1} + \alpha\xi_1, ..., \tau_1 x_{iS} + \alpha\xi_S) \tag{24}$$

so that

$$E[\theta_{ij}|X^*, \xi^{t-1}] = \frac{\tau_1 x_{ij} + \alpha\xi_j}{n_{rep_i} + \alpha} \tag{25}$$

M-step The M-step, which requires the maximization of the expected value of the (log-)likelihood $P(x, \theta|\xi)$ cannot be solved in closed form. For this reason it is necessary to resort to efficient numeric techniques, such as the ones presented by [14] and [15].

5 Results on Simulated data

In this section we will present the results obtained on simulated data set. We will compare the performance of the Naive Bayes (NB) classifier and of the Hierarchical Naive Bayes method with the two different learning approaches herein presented. The simulated data follow a hierarchical structure, since each example has different replicates. To apply the NB approach the n_{rep_i} replicates of the i-th example are summarized by their sample average $\bar{x}_i$. The approaches have been evaluated by computing the classification accuracy and the Brier score calculated with an hold-out strategy.

5.1 Data generation

We generated multiple data sets (simulated experiments). Each experiment included N examples, equally distributed into two classes. Each example had n_{rep} replicates for each attribute. We generated data from n_{feat} independent features. The values of the replicates of the i-th example, for each attribute, have been generated from a gaussian distribution $N \sim (\mu_i, \sigma^2)$ where the individual parameters μ_i have been sampled from a gaussian distribution $N \sim (M, \tau^2)$, with parameters M and τ^2 dependent on the class. The experiments that we have performed can be divided into two subgroups. In subgroup I we have generated 4000 data, 2000 have been used as a training set and 2000 as a test set; in subgroup II we have generated 140 data, 70 of them used for training and 70 for testing. In each subgroup we have then simulated data with 1, 3 and 10 features, with 5 replicates for subgroup I and 3 for subgroup II. For testing the algorithm also with discretized data, we have discretized the obtained data set with 5 bins for each variable.

5.2 Results

The results obtained with the Gaussian model and for the discrete model are shown in Table 1. Accuracy and Brier score are reported with their 95% confidence intervals. The Hierarchical Naive Bayes with empirical learning is denoted with HBN while the Hierarchical Naive Bayes with EM learning is denoted with HBN-EM. Figure 2 shows the ROC curves computed in the Gaussian and discrete cases.

Experiment	Classifier	Accuracy Gaussian data	Brier Score Gaussian data	Accuracy Discrete Data	Brier Score Discrete Data
I-1	HNB	0.928 [0.917-0.938]	0.105	0.860 [0.811-0.909]	0.188
	HBN-EM	0.933 [0.924-0.942]	0.100	0.864 [0.818-0.909]	0.178
	NB	0.874 [0.860-0.889]	0.181	0.841 [0.789-0.893]	0.229
I-3	HNB	0.984 [0.979-0.990]	0.022	0.935 [0.920-0.949]	0.086
	HBN-EM	0.986 [0.981-0.991]	0.021	0.942 [0.927-0.957]	0.076
	NB	0.942 [0.931-0.953]	0.086	0.925 [0.908-0.943]	0.106
I-10	HNB	0.999 [0.997-1.000]	0.002	0.985 [0.980-0.991]	0.019
	HBN-EM	1.000 [0.999-1.000]	0.000	0.989 [0.984-0.994]	0.014
	NB	0.985 [0.980-0.990]	0.023	0.979 [0.973-0.985]	0.031
II-1	HNB	0.889 [0.820-0.958]	0.150	0.849 [0.757-0.940]	0.200
	HBN-EM	0.897 [0.823-0.970]	0.143	0.850 [0.760-0.940]	0.191
	NB	0.864 [0.785-0.944]	0.192	0.842 [0.747-0.938]	0.216
II-3	HNB	0.958 [0.916-1.000]	0.051	0.918 [0.849-0.988]	0.103
	HBN-EM	0.956 [0.912-1.000]	0.050	0.925 [0.862-0.988]	0.094
	NB	0.933 [0.865-1.000]	0.096	0.911 [0.841-0.981]	0.111
II-10	HNB	0.979 [0.958-1.000]	0.021	0.972 [0.943-1.000]	0.030
	HBN-EM	0.992 [0.985-1.000]	0.008	0.959 [0.917-1.000]	0.042
	NB	0.976 [0.951-1.000]	0.028	0.948 [0.897-1.000]	0.052

Table 1. Results obtained on gaussian and discrete data. The accuracy confidence intervals was computed by repeating the data generation, learning and classification steps 100 times for gaussian variables and 60 times for discrete variables.

The two approaches for handling hierarchical data outperforms in both cases the NB one. The results are better in the Gaussian case because the synthetic data that we have generated follow the same distributional assumptions. The HBN with EM learning is always slightly better than HBN with empirical learning and with collapsing, although the difference is minimal in the discrete case. Finally, the advantage of using HBN with respect to NB is statistically significant only in the Gaussian case, in presence of a large data set.

6 Discussion

From the analysis performed on simulated data the proposed HBN approach provides an improvement with respect to the NB. The advantages given by our

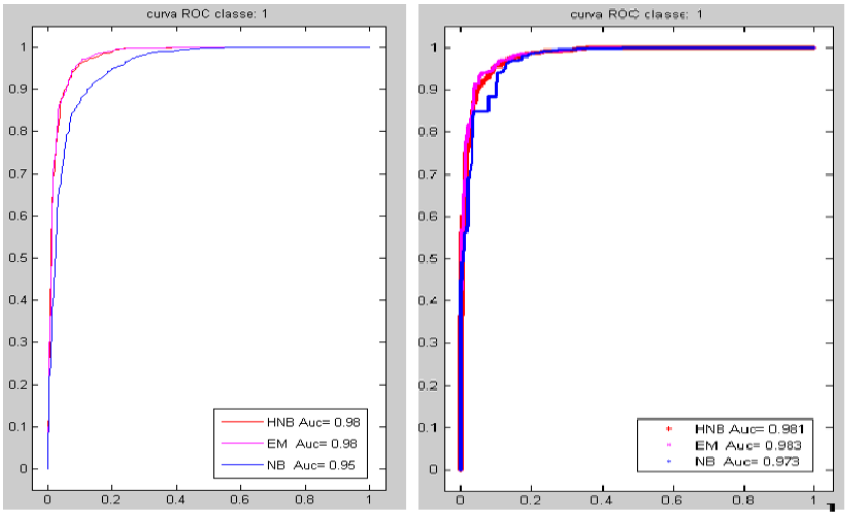

Fig. 2. The ROC curves of experiment 1 for the Gaussian and Discrete cases

algorithm are not dramatic from an accuracy viewpoint, since the improvements in our simulated data set remains limited to the 5%. However, we believe that the proposed methodology should be conveniently applied in presence of repeated measurements at least for two reasons: i) the method is able to take into account the uncertainty in the data in the learning and classification phase; the estimate of the posterior distribution will be closer to the real one and the impact of the evidence will take into account the spread of the replicates. For example, let us suppose that we have collected two sets of three replicates, the first one being $X_1 = [0.0762, 0.1467, 0.1860]$ with average value μ_{X_1}=0.1363 and the second one being $X_2 = [0.0725, 0.1220, 0.1206]$ with average value μ_{X_2}=0.1050. The two replicates are drawn from the same population distribution, with mean equal to zero and and variance equal to one. In the non hierarchical model, the likelihood of the two measurements is very similar, 0.3953 for the first set and 0.3967 for the second set. On the contrary, the computation of the marginal likelihood of the hierarchical one, assuming that both set of measurements are characterized by an individual variance of 0.05, gives 0.6773 and 0.7118 for the two sets, showing that the first set turns out to be clearly less likely than the first one. Such information is used by the learning algorithm, but may be also used during classification to highlight difficult cases or experimental problems. ii) the proposed learning algorithms can be implemented in a rather efficient way; in particular the empirical learning and learning with collapsing strategies do not represent an additional burden with respect to the NB one; moreover, the EM strategy in the Gaussian case is also very efficient, reaching the convergence very fast.

7 Conclusions

The approach proposed in this paper, called Hierarchical Naive Bayes, allows to deal with classification of examples for when repeated measurements are available. It improves the Naive Bayes strategy by avoiding averaging and by properly handling the uncertainty in the data. Our goal is to apply HBN with experimental data; we are currently working on the problem of diagnosing of cancer patients on the basis of Tissue Microarray data [16].

References

1. Bhattacharyya, C., Grate, L.R., Jordan, M.I., El Ghaoui, L., Mian, I.S. Robust sparse hyperplane classifiers: application to uncertain molecular profiling data. J Comput Biol. 11(6) (2004) 1073-89.
2. Ishibuchi, H., Fujioka R., Tanaka, H. Neural Netwroks that learn from Fuzzy if then rules. IEEE Trans. Fuzzy Syst. 1 (2) (1993) 85-97.
3. Domingos, P., Pazzani, M., Provan, G. On the Optimality of the Simple Bayesian Classifier under Zero-One Loss. Machine Learning, 29(2-3) (1997) 103-130.
4. Kononeko, I. Comparison of inductive and naive Bayesian learning approaches to automatic knowledge acquisition. Current Trends in Knowledge Acquisition, IOS Press (1990).
5. Bae, K., Mallick, K. Gene Selection using a two-level hierarchical Bayesian model. Bioinformatics, 20(18) (2004) 3423-30.
6. Cho, H., Lee, J.K. Bayesian hierarchical error model for analysis of gene expression data. Bioinformatics, 20(13) (2005) 2016-25.
7. Hein, A.M.,Richardson, S., Causton, H.C., Ambler, G.K., Green, G.K. BGX a full Bayesian integrated approach to the analysis of affymetrix GeneChip data. Biostatistics, 6(3) (2005) 349-373.
8. Buntine, W.L. Operations for learning with graphical model. JAIR, (2) (1994) 159-225.
9. Langley, P., Thompson, K. An analysis of bayesian classifiers. In: Proceedings of the tenth national conference on artificial intelligence, AAAI Press and MIT Press (1992).
10. Gelman, A., Carlin, J. B., Stern, H.S., Rubin, D.B. Bayesian data analysis. Chapman & Hall (1997).
11. Leonard, T. Bayesian simultaneous estimation for several multinomial experiments. Commun. Statist. Theor. Meth., A6(7) (1977) 619-630.
12. Bellazzi, R., Riva, A. Learning Bayesian Networks probabilities from longitudinal data. IEEE transactions on systems, man and cybernetics, 28(5) (1998) 629-636.
13. Minka, T.P. Estimating a Dirichlet distribution, Tutorial published on the web at: 'http://research.microsoft.com/ minka/papers/dirichlet/' (2003).
14. Ronning, G. Maximum Likelihood estimation of Dirichlet distribution. Journal of statistical computation and simulation 32(4) (1989) 215-221.
15. Naryanan, A. Algorithm as 266: Maximum Likelihood estimation of the parameters of the dirichlet distribution. Applied Statistics 40 (1991) 365-374.
16. Demichelis, F. On Information Organization and Information Extraction for the Study of Gene Expressions by Tissue Microarray Technique. PhD Thesis, International Doctorate School in Information and Communication Technologies, University of Trento, Italy (2005).