Attribute Relevance in Multiclass Data Sets Using the Naive Bayes Rule*

J.M. Sotoca, J.S. Sánchez, F. Pla
Dept. Llenguatges i Sistemes Informátics. Universitat Jaume I.
Av. Sos Baynat s/n, E-12071 Castellón, Spain
(sotoca,sanchez,pla)@lsi.uji.es

Abstract

Feature selection using the naive Bayes rule is presented for the case of multiclass data sets. In this paper, the EM algorithm is applied to each class projected over the features in order to obtain an estimation of the class probability density function. A matrix of weights per class and feature is then obtained, where it collects the level of relevance of each feature for the different classes. We show different ways to extract this information and compare the behavior of the ranking of relevance obtained applying the naive Bayes and K-NN classifiers.

1. Introduction

In the establishment of a pattern recognition system, there appear situations where the class conditional distribution of the system is unknown and only a training set (TS) of labeled patterns is available. Apart from the rather trivial cases where information is governed by a simple distribution which can be found from it, it is necessary to obtain an estimator that collects the different class-conditional probability density functions (pdf). This estimation is important when we want an accuracy measure of the divergence or dissimilarity between classes to establish a feature ranking and to evaluate the effectiveness of class discrimination to be used in *feature selection*.

There are two ways of addressing the problem of *feature selection*. One of them consists of creating new features by transforming or combining the initial features. This supposes a transformation of the feature space and correspondingly, a change over the training patterns. The other trend allows to select a subset of the original features in order to reduce the feature space dimensionality without a significant decrease in classification performance.

*This work has been supported by grants TIC2003-08496-C04-03 and CPI2001-2956-C02-02(Spanish CICYT), P1-1B2002-07(Fundació Caixa Castelló-Bancaixa), and IST-2001-37306 (IST Project European Union).

In this work, we address the *feature selection* problem in the sense of dimensionality reduction. To this end, we apply the naive Bayes classifier in multiclass data sets to obtain the order of attribute relevance for each class. In practice, the class distribution cannot be represented by an unimodal *pdf*, thus requiring a mixture of modes. The *expectation-maximization* (EM) algorithm [2] is here used to estimate a mixture of modes for each class projected over the features. Afterwards, the mixture of modes will be used to determine the class separability in each feature. A matrix of weights per class and feature is obtained, collecting the relevance level of each feature for the different classes.

2. Feature discrimination power per class

In general, we have a TS with N_T instances, where each instance $\mathbf{x}$ is a point $\mathbf{x} = \{x_1, \ldots, x_d\} \in \Re^d$, in a d-dimensional feature space and $C = \{c_1, \ldots, c_J\}$ is the set of the different classes present. This instances follow a spatial class distribution according to their true class-conditional *pdf* $P(\mathbf{x}|c_j)$ and the respective *a priori* probability $P(c_j)$, $c_j \in C$. Thus we can say that a point $\mathbf{x}$ can be then optimally classified using the Bayes rule based on the knowledge of the components $P(c_j)P(\mathbf{x}|c_j)$ for each class c_j. A simplification using naive Bayes rule consists of assuming that the features must be statistically independent. This independence is defined by means of the product of the probabilities in each feature $p(\mathbf{x}) = \prod_{k=1}^{d} p(x_k)$ as follows:

$$C_{Naive} = \max_{j=1\ldots c_J} P(c_j) \prod_{k=1}^{d} f^*(x_k|c_j) \qquad (1)$$

where $f^*(x_k|c_j)$ is an estimation of class-conditional *pdf* of the class c_j in the feature k. A natural way to deal with a density estimator is to consider a mixture of modes:

$$f^*(x_k|c_j) = \sum_{m=1}^{M} \alpha_{m|kj} \Theta_{m|kj}(x_k) \qquad (2)$$

where $\Theta_{m|kj}$ is a class-conditional *pdf* and $\alpha_{m|kj}$ is the *a priori* probability of the mode m^{th} in the projection of class c_j over feature k, such that $\sum_{m=1}^{M} \alpha_{m|kj} = 1$. Each mode can be characterized by means of a normal distribution, where $\mu_{m|kj}$ and $\sigma_{m|kj}$ are its mean and standard deviation, respectively. In order to optimize the accuracy of the naive Bayes rule, the EM algorithm is used to find the conditional *pdf* for the different classes.

2.1. Spatial class distribution for each feature

In general, the EM algorithm for fitting finite mixtures has several drawbacks [3]: it is a local method, thus sensitive to initialization because the likelihood function of a mixture model is not unimodal. To solve this problem, in the initial stage the mixture of gaussians is placed uniformly along the frequencies the class projected over each feature. Another important issue in the model refers to the selection of the number of components: with too many components the mixture may overfit the data, while with too few components it may not be flexible enough to approximate the true underlying model.

In order to adjust the distribution to a mixture of m gaussians, we iterate the following two steps:

E-step Compute the conditional *pdf* of prototype $x_k^t \in \{x_k^1, \ldots, x_k^{N_j}\}$ over the mode m^{th}, where N_j is the number of elements from the class c_j projected over the feature k.

$$\Theta_{m|kj} = \frac{\alpha_{m|kj} N(x_k^t : \mu_{m|kj}, \sigma_{m|kj})}{\sum_{l=1}^{M} \alpha_{l|kj} N(x_k^t : \mu_{l|kj}, \sigma_{l|kj})} \quad (3)$$

M-step Compute the parameters of the modes. $h(x_k^t)$ represents the frequency of each value x_k^t in class c_j projected over the feature k.

$$\alpha_{m|kj} = \frac{1}{N_j} \sum_{t=1}^{N_j} h(x_k^t) \Theta_{m|kj} \quad (4)$$

$$\mu_{m|kj} = \frac{\sum_{t=1}^{N_j} h(x_k^t) \Theta_{m|kj} x_k^t}{\sum_{t=1}^{N_j} h(x_k^t) \Theta_{m|kj}} \quad (5)$$

$$\sigma_{m|kj} = \frac{\sum_{t=1}^{N_j} h(x_k^t) \Theta_{m|kj} (x_k^t - \mu_{m|kj})^2}{\sum_{t=1}^{N_j} h(x_k^t) \Theta_{m|kj}} \quad (6)$$

The algorithm stops at the iteration N when the following conditions are fulfilled over all modes:

$$\mu_{m|kj}^{N} - \mu_{m|kj}^{N-1} < 0.1 * \left(\max(x_k^t) - \min(x_k^t)\right)$$

$$\sigma_{m|kj}^{N} - \sigma_{m|kj}^{N-1} < 0.05 * \left(\max(x_k^t) - \min(x_k^t)\right)$$

where $\max(x_k^t)$ and $\min(x_k^t)$ are the maximum and minimum values of the class c_j projected over the feature k.

2.2 Discrimination power per class and feature

Motivated by the need of improving the performance of the *feature selection* process, several works have employed a measure of divergence to find a good feature subset [6][1]. Here we apply the divergence or separability between classes using the Kullback-Leibler distance. For a set of classes, if a class c_j is picked and the rest of classes are joined to form a unique class, say c_{J-j}, then the discrimination power of each feature k for class c_j can be expressed as follows:

$$KL(k|c_j, c_{J-j}) = \sum_{t=1}^{N_T} f^*(x_k^t|c_j) log \frac{f^*(x_k^t|c_j)}{f^*(x_k^t|c_{J-j})} \quad (7)$$

Due to asymmetry of Kullback-Leibler distance, it is necessary to employ a certain transformation of it in order to obtain a symmetric measure of divergence:

$$D(k|c_j) = KL(k|c_j, c_{J-j}) + KL(k|c_{J-j}, c_j) \quad (8)$$

Using this measure, for each feature we have a set of weights per class. The most common way to obtain a measure of relevance is considering the average over all classes:

$$D(k) = \frac{1}{J} \sum_{j=1 \ldots c_J} D(k|c_j) \quad (9)$$

On the other hand, there may exist a feature relevant to one class (i.e., it has high divergence for that class), but being not relevant for the other classes. Then, the choice of a feature subset has to collect those attributes that indeed separate certain classes. To pick up this information, we normalize the set of weights for all features obtaining the follow measure:

$$Dnorm(k) = \frac{1}{J} \sum_{j=1 \ldots c_J} \frac{D(k|c_j)}{\sum_{t=1}^{d} D(t|c_j)} \quad (10)$$

In some cases, it is possible to obtain an improvement in the ranking of relevance making a search in the matrix of weights per classes and features. Using the normalized weights of the previous expression, sort the weights for each class such the feature appearing in the first place is that with the highest discrimination power. One solution is to pick up the features found in the first places of the matrix. Nevertheless, there exist connections between features and it is not clear how to obtain an optimal choice of the internal order of relevance in the feature subset.

The solution proposed in this paper consists of applying a Sequential Backward Floating Selection (SBFS) scheme, where in each iteration we eliminate the worst feature and reduce the number of columns in the matrix following a criterion of averaging the normalized weights by the a priori

SBFS scheme in the matrix of weights

$|X| = d$; //Set of features
$col = d$; //number of columns
EXCLUSION:$X^- \leftarrow argmaxF(|X| - x_k)$;
$|X| - x_k; x_{loc} = x_k$; // New subset of features
$col = col - 1$;
while do
  1) EXCLUSION:$X^- \leftarrow argmaxF(|X| - x_k)$;
  $|X| - x_k; x_{back} = x_k$; // New subset of features
  2) INCLUSION:$X^+ \leftarrow argmaxF(|X| \cup x_k)$;
  $x_{for} = x_k$;
  if $(x_{loc} \neq x_{for})$ **then**
    $|X| \cup x_{for}$; // New subset of features
  end if
  $x_{loc} = x_{back}$;
  if $(col > 1)$ **then**
    $col = col - 1$;
  end if
end while$|X| \neq 0$

probability of that class:

$$F = \sum_{k=1}^{d} \frac{1}{J} \sum_{j=1 \ldots c_J} P(c_j) \frac{D(k|c_j)}{\sum_{t=1}^{d} D(t|c_j)} \qquad (11)$$

This algorithm allows to recover features eliminated in previous iterations through a conditional inclusion. Thus a feature eliminated in previous steps can be relevant when the number of columns utilized in the matrix is lower. The subset of features that survive when the algorithm arrives at the first column is a suboptimal subset that collects the most relevant features. Finally, we establish the internal order of relevance for the features in the subset following the criterion of Eq. 11.

Table 1. Characteristics of the data sets.

	Features	Classes	Instances
Waveform21	21	3	5000
Waveform40	40 (19)	3	5000
Images	19	7	2310
Letter	16	26	20000
Texture	40	11	5500
Diabetes	8	2	768
Glass	9	6	214
Iris	4	3	151
Pendigits	16	10	10992
Vowel	10	10	528
Vehicle	18	4	848
Wine	13	3	178

3 Empirical results

In this section, we compare the behavior of different rankings of relevance obtained with the methods previously described and *ReliefF*. The *ReliefF* is an extension to the classical algorithm *Relief* adapted by Kononenko [4]. Quality of rankings has been evaluated with the naive Bayes and K-Nearest Neighbor (K-NN) rules. In the case of the K-NN rule, the normalized Euclidean distance is used.

From the UCI Repository [5], twelve artificial and real data sets have been selected. These data sets present different degrees of attribute relevance along with irrelevant features. The main characteristics of the data sets are summarized in Table 1 (the number of irrelevant features are given in brackets). To increase statistical significance of the results in domains with a limited number of instances, 5-fold cross-validation has been employed.

To measure quality of the ranking obtained by the different filter methods, a SFS (Sequential Forward Selection) scheme is performed from the best subset of relevant features. Table 2 and Table 3 provide the classification accuracy of the best subset found for the different filter methods and its dimensionality, using the naive Bayes and K-NN rules, respectively. The best mixture of modes and value of K-NN are chosen when all features are presented (with a number of gaussians from 1 through 7 and K ranging from 1 through 21).

From the results reported in Table 2 and Table 3, it can be noted two situations. First, in the case of data sets where there exist important differences in the level of attribute relevance between classes (Waveform21, Waveform40, Images, Vehicle), the filter methods that obtain a matrix of weights per feature and class can recover this information and improve the results of the feature subsets. On the other hand, in datasets where the degree of relevance for each class is similar (Letter, Pendigits, Vowel), this advantage is losed with respect to the *ReliefF*. Finally, Glass data set is a special case in which accuracy obtained by *ReliefF* is clearly superior to that of the rest. The number of points in this data set is too small and with an irregular distribution, and it is necessary to use a higher number of modes to obtain an appropriate representation.

4 Conclusions

In this paper, a *feature selection* method based on the naive Bayes rule is presented for the case of multiclass data sets. A matrix of weights per class and feature is obtained making a study of the attribute relevance in each class.

Normalized weights are used to collect the level of the attribute relevance per each class and search for those features that indeed are important in the classification accuracy to

	#	D(k)		Dnorm(k)		SBFS			ReliefF	
	gauss.	% class.	dim.	% class.	dim.	% class.	dim.	Concl.	% class.	dim.
Waveform21	2	81.45±1.20	17.4	81.87±1.38	13.4	**81.93±1.37**	15.4	$>$	81.49±1.21	15.6
Waveform40	5	81.01±0.46	14	**82.11±0.80**	10.4	**82.11±0.80**	10.4	$>$	81.59±0.69	13
Images	5	91.81±2.78	10.6	91.81±2.78	10.2	**92.16±2.38**	9.2	$>$	91.86±2.42	16.2
Letter	2	71.70±0.27	13.2	71.66±0.27	14.8	71.70±0.27	13.2	$=$	71.77±0.31	14.2
Texture	7	80.05±0.30	16.8	80.61±0.33	16.8	80.65±0.69	16.4	$=$	80.61±0.45	11.6
Diabetes	1	76.69±1.41	5.2	76.69±1.41	5.2	**77.34±1.36**	5.2	$>$	76.68±1.80	5.8
Glass	5	63.99±6.04	7.4	64.45±5.04	7.6	64.45±5.06	7.6	$<$	**68.78±7.23**	6.8
Iris	7	98.66±1.82	2	98.66±1.82	2.6	98.66±1.82	2.6	$=$	98.66±1.82	2.8
Pendigits	1	84.27±0.55	16	84.27±0.55	16	84.27±0.55	16	$<$	**84.61±0.53**	15
Vowel	7	77.39±10.5	8.8	77.39±10.5	8.8	77.39±10.5	9.4	$=$	77.39±10.5	10
Vehicle	4	**63.45±3.84**	10.6	63.33±3.80	13.6	63.15±3.77	16.4	$>$	62.90±3.07	17.4
Wine	3	99.42±1.27	5.8	99.42±1.27	6	99.42±1.27	7.2	$=$	99.42±1.27	8.2

Table 2. Accuracy (% class.) of the naive Bayes classifier with different filter methods and size (dim.) of the best feature subset selected.

	#	D(k)		Dnorm(k)		SBFS			ReliefF	
	K-NN	% class.	dim.	% class.	dim.	% class.	dim.	Concl.	% class.	dim.
Waveform21	21	85.62±1.02	19.2	85.62±1.02	18.4	85.51±1.16	18.6	$=$	85.51±1.16	18.6
Waveform40	21	86.19±1.19	15.4	86.31±1.46	18.6	86.21±1.16	15.2	$<$	**86.83±1.06**	20
Images	1	97.31±1.68	12.6	97.36±1.72	13.2	97.44±1.81	13	$=$	97.44±1.63	14.6
Letter	5	95.11±0.11	11	95.11±0.11	11	95.11±0.11	11	$=$	95.11±0.11	11
Texture	1	99.18±0.28	31.2	99.18±0.28	31.8	99.16±0.22	34.6	$=$	99.23±0.27	32.6
Diabetes	17	76.69±1.95	4.8	76.69±1.95	3.6	**77.60±2.54**	4.2	$>$	75.65±1.62	7
Glass	1	72.90±4.75	5.6	72.96±4.74	5.8	73.39±5.44	5.8	$<$	**74.09±7.30**	4.2
Iris	1	99.33±1.48	2.6	99.33±1.48	2.6	99.33±1.48	2.6	$=$	99.33±1.48	2.8
Pendigits	3	99.05±0.00	15.8	99.03±0.04	16	99.03±0.04	16	$=$	99.03±0.04	16
Vowel	1	98.08±1.10	9	98.08±1.10	9	**98.26±1.19**	9	$>$	97.91±0.98	9.6
Vehicle	7	71.37±2.46	13.2	72.44±2.76	16	**72.56±2.80**	15.2	$>$	69.38±2.30	15.6
Wine	3	98.84±1.58	6.6	**98.84±1.58**	6	98.84±1.58	6.6	$>$	98.26±2.57	9

Table 3. Accuracy (% class.) of the K-NN classifier with different filter methods and size (dim.) of the best feature subset selected.

certain classes. This information allows to obtain better internal order of relevance when there exist more differences among the various classes. In these data sets, we improve the results of the best subset selected with respect to that of the well-known ReliefF approach.

References

[1] M. Bressan and J. Vitriá. Independent component analysis and naive bayes classification. In *Proc. of 2nd Conf. on Visualization, Imaging and Image Processing*, pages 496–501, Benalmadena, (Spain), 2002.

[2] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *J. Royal Stat. Soc.*, 39:1–38, 1977.

[3] M. A. T. Figueiredo and A. K. Jain. Unsupervised learning of finite mixture models. *IEEE Trans. on PAMI*, 24:381–396, 2002.

[4] I. Kononenko. Estimating attributes: analysis and extensions of relief. In *Proc. of 7th European Conf. on Machine Learning*, pages 171–182, Catania, Italy: Springer Verlag, 1994.

[5] P. M. Murphy. Uci repository of machine learning. http://www.ics.uci.edu/AI/ML/MLDBRepository.html, 1995. Department of Information and Computer Science, University of California, Irvine, CA.

[6] J. Novovicová, P. Pudil, and J. Kittler. Divergence based feature selection for multimodal class densities. *IEEE Trans. on PAMI*, 18:218–223, 1996.