Feature Selection Based on a New Formulation of the Minimal-Redundancy-Maximal-Relevance Criterion

Daniel Ponsa and Antonio López

Centre de Visió per Computador, Universitat Autònoma de Barcelona
Edifici O, 08193 Bellaterra, Barcelona, Spain
{daniel,antonio}@cvc.uab.es
www.cvc.uab.es/adas

Abstract. This paper proposes an incremental method for feature selection, aimed at identifying attributes in a dataset that allow to buid *good* classifiers at low computational cost. The basis of the approach is the minimal-redundancy-maximal-relevance (mRMR) framework, which attempts to select features relevant for a given classification task, avoiding redundancy among them. Relevance and redundancy have been popularly defined in terms of information theory concepts. In this paper a modification of the mRMR framework is proposed, based on a more proper quantification of the redundancy among features. Experimental work on discrete–valued datasets shows that classifiers built using features selected by the proposed method are more accurate than the ones obtained using original mRMR features.

1 Introduction

In the last years, many research efforts in areas like machine learning, clustering and data mining have focused on problems that require the management of big volumes of data. Applications like multimedia indexing, text classification or gene expression array analysis demand these disciplines to work in domains with tens or hundreds of thousands of variables, while domains hardly surpassed the hundred of variables just few years ago. Elements in datasets are described by a huge number of variables, and selecting a relevant subset of them upon which to focus attention is still an open problem. This is the task carried out by variable or feature selection algorithms[1]. The objective pursued when selecting features may depend on the final application, being common purposes:

- removing useless features (noise or distracters) in order to save computing time and data storage;
- improving the performance of a predictor (a classifier or a regressor) learned from the dataset, since the risk of over-fitting decreases if the domain of the problem has lower dimensionality;

[1] In this paper we use equivalently the terms variables and features to denote the attributes of elements in a dataset.

J. Martí et al. (Eds.): IbPRIA 2007, Part I, LNCS 4477, pp. 47–54, 2007.

– making the understanding of the process that has generated the dataset easier (reverse engineering).

For an excellent introduction on feature selection algorithms, the reader is referred to [1], where different contributions in this field are compiled.

The topic of this paper is feature selection for pattern classification. Given a labeled dataset of N samples $\{(\mathbf{x}_i, l_i)\}_{i=1}^{N}$, where $\mathbf{x}_i = \{x_{ij}\}_{j=1}^{M}$ is a vector of M features, and l_i the target classification label, the feature selection problem consists in finding, from the observation space $\mathbb{R}^M$, a subspace of at most $m \ll M$ features $\mathbb{R}^m$ from where to train a classifier characterizing l. The difficulty of this task is that the number of possible subspaces in $\mathbb{R}^M$ of dimension less or equal to m is $\sum_{i=1}^{m} \binom{M}{m}$, and examining all of them exhaustively can not be done in practice. To deal with this problem, different algorithms have been proposed, based on suboptimal strategies to explore the space of solutions (greedy search, best–first, genetic algorithms, etc.). These methods require a criterion to evaluate the inspected subsets and guide the search. Two different philosophies can be distinguished depending on the nature of this criterion, namely *wrapping* and *filtering*.

Wrappers [2] use the learning machine of interest (the one to be used in the final classifier) to score subsets according to their predictive power, selecting in that way features tailored to a particular algorithm. On the other hand, filters evaluate subsets by criteria independent of the chosen predictor. In general, filters have a lower computational cost than wrappers, and are commonly used to preprocess datasets before training classifiers, or before starting a wrapping process.

This paper proposes a filtering method based on the *minimal-redundancy-maximal-relevance* (mRMR) proposal in [3]. The aim of the mRMR approach is to select a subset of features well–suited for a given classification task, by taking into account the relevance (ability) of features to identify the classification label, as well as the redundancy among them. These concepts are defined in terms of the Mutual Information between features. This paper proposes a modification of the (mRMR) proposal, based on an alternative formulation of the redundancy among features. The significance of this modification is evaluated experimentally, by comparing the performance of classifiers in two different problems, when features selected by the original mRMR method or the ones selected by the proposed method are used.

This paper is organized as follows. Section 2 reviews information theory concepts necessary to understand the mRMR criterion, which is presented in section 3. Then, section 4 proposes modifications on the mRMR criterion, to make a more coherent formulation of the concept of redundancy among features. Section 5 quantifies the benefits of the proposed method, by comparing experimentally the accuracy of classifiers trained with features selected by the presented proposal, against the accuracy of classifiers trained with original mRMR features. The paper ends by drawing some conclusions.

2 Information Theory Concepts

Consider a discrete[2] random variable x, taking values in the set of N symbols $\{s_i\}_{i=1}^{N}$. Let $P(s_k)$ be the probability of $x = s_k$, then, the (Shannon's) information obtained when we are informed that $x = s_k$ is computed from the expression

$$I(x) = \log\left(\frac{1}{P(x)}\right) = -\log\left(P(x)\right) \quad . \tag{1}$$

Thus, the information received by observing $x = s_k$ is bigger the less likely s_k is. If the basis 2 is used for the logarithm, the information in x is measured in bits. For example, if x is a binary variable with states $\{s_1, s_2\}$ of equal probability (i.e. $P(s_1) = P(s_2) = 0.5$), a bit of information is obtained when the state of x is observed. On the other hand, if $P(s_1) = 1$, then the observation of the value of x does not really provide information, since it is already known that its value will be s_1. Equation (1) quantifies the information obtained when a *concrete* value of x is observed. The information that the variable x can provide corresponds to the average of the information provided by the different values it may take. This corresponds to the expectation of the information $E[I(x)]$, which is computed by

$$H(x) = \sum_{i=1}^{N} P(s_i)I(s_i) = -\sum_{i=1}^{N} P(s_i)\log\left(P(s_i)\right) \quad . \tag{2}$$

By $H(x)$ is denoted the *Shannon's entropy* of x, and it is a measure of its uncertainty. Using this same concept, one can compute the uncertainty about x when the state of a second discrete random variable $y = s_j$ is known. Similarly to expression (2), the *conditional entropy* of x given $y = s_j$ is computed by

$$H(x|s_j) = -\sum_{i=1}^{N} P(s_i|s_j)\log\left(P(s_i|s_j)\right) \quad .$$

The conditional entropy of x for any value of y corresponds to the weighted average (with respect to $P(y)$ probabilities) of the entropies $H(x|s_j)$. Therefore, it corresponds to

$$H(x|y) = -\sum_{j=1}^{N} P(s_j)\sum_{i=1}^{N} P(s_i|s_j)\log\left(P(s_i|s_j)\right) \quad .$$

$H(x|y)$ measures the uncertainty in x once variable y is known. Combining $H(x|y)$ and $H(x)$, one can quantify the information about x provided by y, what has been termed as the *mutual information* between x and y. This is done with the expression

$$MI(x, y) = H(x) - H(x|y) \quad .$$

[2] Continuous variables imply equivalent expressions, replacing summatories by integrals, and probabilities by density functions.

That is to say, if from the uncertainty of x one subtracts the uncertainty of x once y is known, this provides the information (in bits) that variable y provides about x. MI is widely used as a measure to determine the dependency of variables. It can be shown that MI can also be computed by

$$MI(x,y) = \sum_{i=1}^{N}\sum_{j=1}^{N} P(s_i,s_j)\log\frac{P(s_i,s_j)}{P(s_i)P(s_j)} \ .$$

Next section describes a proposal to use this concept in order to select a proper subset of features for classification tasks.

3 mRMR Criterion

Many filter approaches to feature selection are based on first ranking features according to a scoring function, to then defining a subset of them from the m ones of highest score. For classification tasks, this scoring function has to identify features with the highest relevance to the class label l. One common approach to quantify this relevance is the mutual information between features and the class label $MI(x,l)$.

The major weakness of this scheme is that, in order to obtain a good classification performance, selecting the m best features does not imply that the best subset of m features is actually selected. Indeed, the combination of two *good* features does not imply improving the classification performance, if both have a very similar behavior in determining l. Thus, it seems reasonable to avoid the *redundancy* among selected features. The work in [3] proposes to do that with the heuristic *minimal–redundancy–maximal–relevance* framework, where the redundancy between two features x and y is determined by $MI(x,y)$. The task of feature selection is posed as selecting from the complete set of features S, a subset S_m of m features that maximizes

$$\frac{1}{m}\sum_{x_i\in S_m} MI(x_i,l) - \frac{1}{\binom{m}{2}}\sum_{x_i,x_j\in S_m} MI(x_i,x_j) \ . \tag{3}$$

This expression takes into account the relevance of features with the class label while penalizing redundancy among them. Since the search space of subsets of m elements in $\mathbb{R}^M$ is too big to be explored in practice, the paper proposes to determine S_m incrementally by means of a forward search algorithm. Having a subset S_{m-1} of $m-1$ features, the feature $x_i \in \{S - S_{m-1}\}$ that determines a subset $\{x_i, S_{m-1}\}$ maximizing (3) is added. It can be shown that this nested subset strategy is equivalent to iteratively optimize the following condition:

$$\max_{x_i\in S-S_{m-1}} \left(MI(x_i,l) - \frac{1}{m-1}\sum_{x_j\in S_{m-1}} MI(x_j,x_i) \right) \ . \tag{4}$$

Experiments in [3] show that for subsets of more than 20 features, the S_m obtained with this method achieves more accurate classification performances

than the subset obtained by maximizing the $MI(S_m, l)$ value[3] , while the required computation cost is significantly lower.

4 A Reformulation of the mRMR Criterion

The criterion presented in the previous section is based on an heuristic formulation of how a subset of *good* features for classification purposes should be. It should contain features that provide maximal information of the classification label l, while providing minimal information of the rest of the features in the set. These two concepts are considered in (4) by quantifying this information in bits. However, we found that in order to quantify the redundancy among features, this is not a proper way to proceed. Let's consider an example where $MI(x, y) = 0.5$. This half a bit of information may correspond to different redundancy situations of x and y. For instance, if the uncertainty (information) in x is equal to 0.5 (i.e., its entropy is $H(x) = 0.5$), then $MI(x, y) = 0.5$ tells that y provides at least the same information than x. This means that if a feature subset S_m already contains y, adding the feature x is completely redundant. On the other hand, if $H(x) > 0.5$ and $H(y) > 0.5$, then the redundancy among both variables is *weaker*. By using expression (4) to select features, these two situations are considered identical. In order to represent the concept of redundancy more properly, this paper proposes the use of the coefficient of uncertainty [4]. Given two variables x and y, the term

$$R(x, y) = \frac{MI(x, y)}{H(y)} \quad , \tag{5}$$

quantifies the redundancy of y with respect to x with a value between $[0, 1]$, with 1 being a situation of complete redundancy. It measures the ratio of information of y already provided by x. Note that this definition of redundancy is non–symmetric (i.e., $R(x, y)$ may not be equal to $R(y, x)$), which is consistent with reality. The fact that the value of x can completely determine the value of y does not mean that the same occurs in the opposite direction. Examples are feature relationships like $y = |x|$, or $y = x^2$. In these cases, adding x in a set that contains y is not completely redundant (some new information is added to the data set), while the opposite assignment is uninformative. By using the redundancy factor in (5), now a new incremental feature selection algorithm is proposed, based on maximizing at each iteration the condition

$$\max_{x_i \in X - S_{m-1}} \left(MI(x_i, l) - \frac{1}{m-1} \sum_{x_j \in S_{m-1}} R(x_j, x_i) \right) \quad .$$

[3] That is, the mutual information between the whole subset of variables and the classification label l.

5 Experiments

In order to test the proposed feature selection algorithm, the experiments on discrete data proposed in [3] have been reproduced, using the Multiple Features dataset of handwritten digits (HDR) and the Arrhythmia dataset (ARR). Both datasets, available at the UCI repository [5], describe its elements by a relatively big number of continuous features. In the experiments, features have been discretized as proposed in [3]. Table 1 details dataset characteristics.

Table 1. Datasets used in the experiments

Dataset	Acronym	Discretization	# Classes	# Examples/class	# Features
Multi-Feature	HDR	-1 if $x_i \leq \mu$ 1 otherwise	10	200 per class	649
Arrhythmia	ARR	-1 if $x_i < \mu - \sigma$ 1 if $x_i > \mu + \sigma$ 0 otherwise	2	237 and 183	278

Performance of the mRMR and the proposed method is evaluated by a 10-fold cross validation procedure (Figure 1). From each dataset, 10 pairs of training/testing sets are generated. Then, the feature selection algorithm is applied in each training set, generating a sequence of 50 nested subsets $S_1 \subset S_2 \subset \ldots \subset S_{50}$ maintaining the features selected to solve the classification task. For a given subset S_i, a linear Support Vector Machine (SVM) is trained using the training portion of the dataset, and then its accuracy is quantified by the classification error rate on the testing portion. To train and test classifiers, the SVM implementation in the LIBSVM package [6] has been used, using default parameters.

The feature selection methods implemented require the availability of the probability terms $P(x)$ and $P(x, y)$ for features in the dataset. These terms

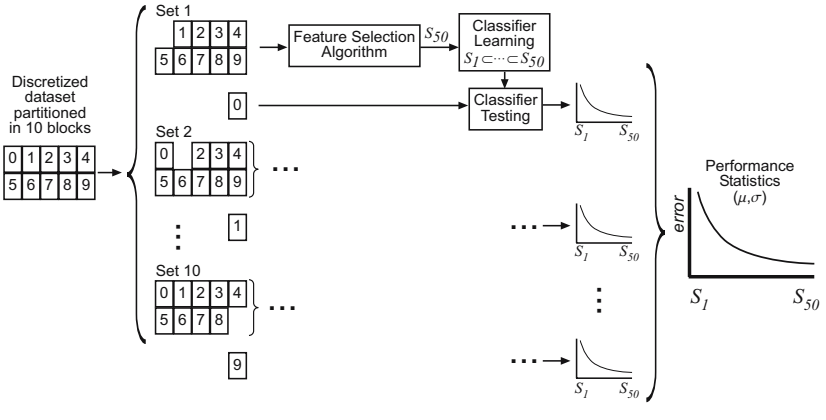

Fig. 1. 10–fold Cross-Validation process

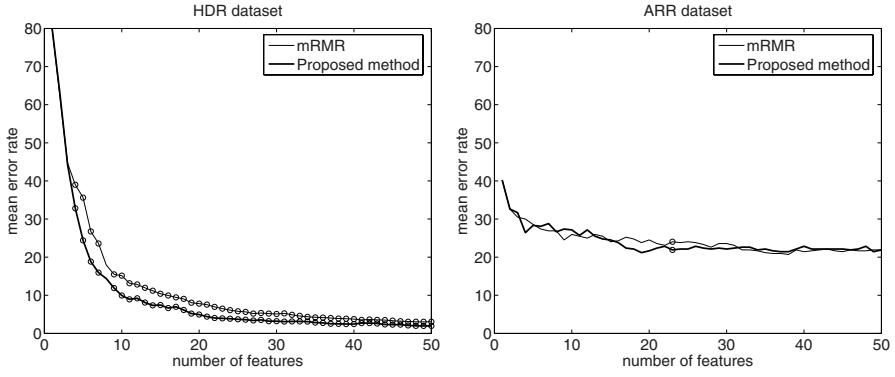

Fig. 2. Mean error rate of the classifiers learned using the mRMR approach, and the proposed method, for the HDR and ARR datasets. Circles on plotted lines highlight cases where the difference in performance is statistically significant ($\alpha = 0.05$).

have been simply estimated by counting the relative frequency of the categorical symbols of each feature in the dataset. For instance, $P(x = 1)$ is determined by the number of examples in the dataset having $x = 1$, divided by the total number of examples.

Figure 2 shows, for each dataset, the mean classification error rate of the classifiers learned using features selected by the two methods considered. For each subset of features S_i, an statistical test has been applied to determine whether the classifiers obtained with the proposed method really outperform the mRMR ones. This has been done using the 10-Fold Cross-Validated Paired t test described in [7]. For the cases in which methods have a statistical significant difference in their performance, plots have been highlighted with a circle.

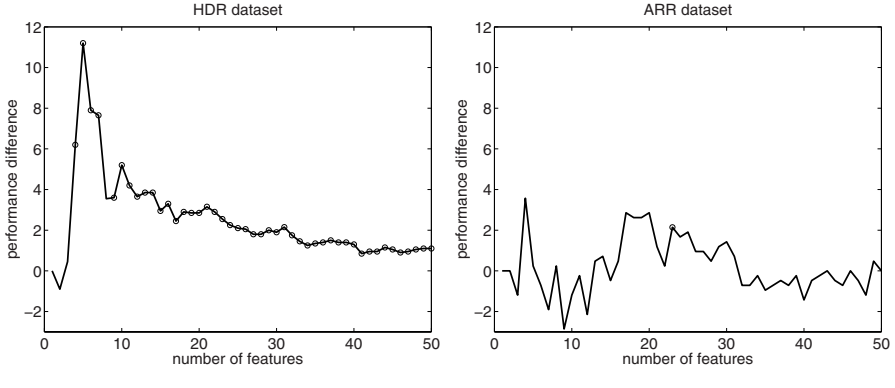

Fig. 3. Difference between the mean error rates obtained with each method. Positive values correspond to improvements provided by the proposed method. Circles on plotted lines highlight statistically significant differences ($\alpha = 0.05$).

Results show that in the HDR dataset, the performance achieved when using the proposed method is clearly better for most of the subsets of features considered. For the ARR dataset, both methods perform similarly. Only in one case one method significantly outperforms the other, and this method is the one proposed in the paper. For a clearer view of the improvement achieved in the average classification performance, Figure 3 shows the difference between the mean error rate obtained with each method.

6 Conclusions

A feature selection method for classification tasks has been presented. The proposal is based on the mRMR framework [3], an introduces an alternative criterion to quantify the redundancy among features. This criterion leads to a feature selection algorithm better suited for classification purposes than the original mRMR proposal. This has been certified in experiments done, measuring the performance of SVM classifiers when trained using the features selected by our proposal and the ones by [3]. Results show that with the features selected by the proposed method, classifiers perform at least as well, and in many cases statistically significantly better, than using the original mRMR selected features.

Acknowledgments. This research has been partially funded by Spanish MEC project TRA2004-06702/AUT.

References

1. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research (3), 1157–1182 (2003)
2. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artificial Intelligence 97(1–2), 273–324 (1997)
3. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: Criteria of max–dependency, max–relevance, and min–redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(8), 1226–1238 (2005)
4. Press, W.H., Flannery, B., Teukolsky, S.A., Vetterling, W.T.: Numerical recipes in c. Cambridge (1988)
5. Newman, D.J., Hettich, S., Blake, C., Merz, C.: UCI repository of machine learning databases (1998)
6. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. (2001)
7. Dietterich, T.G.: Approximate statistical test for comparing supervised classification learning algorithms. Neural Computation 10(7), 1895–1923 (1998)