Feature selection, mutual information, and the classification of high-dimensional patterns: Applications to image classification and microarray data analysis

Article *in* Pattern Analysis and Applications · August 2008

DOI: 10.1007/s10044-008-0107-0 · Source: DBLP

CITATIONS
67

READS
525

3 authors, including:

F. Escolano
University of Alicante
133 PUBLICATIONS **1,385** CITATIONS

SEE PROFILE

Miguel Cazorla
University of Alicante
220 PUBLICATIONS **1,563** CITATIONS

SEE PROFILE

Pattern Anal Applic (2008) 11:309–319
DOI 10.1007/s10044-008-0107-0

THEORETICAL ADVANCES

Feature selection, mutual information, and the classification of high-dimensional patterns

Applications to image classification and microarray data analysis

Boyan Bonev · Francisco Escolano · Miguel Cazorla

Received: 6 January 2007 / Accepted: 20 January 2008 / Published online: 22 February 2008

Abstract We propose a novel feature selection filter for supervised learning, which relies on the efficient estimation of the mutual information between a high-dimensional set of features and the classes. We bypass the estimation of the probability density function with the aid of the entropic-graphs approximation of Rényi entropy, and the subsequent approximation of the Shannon entropy. Thus, the complexity does not depend on the number of dimensions but on the number of patterns/samples, and the curse of dimensionality is circumvented. We show that it is then possible to outperform algorithms which individually rank features, as well as a greedy algorithm based on the maximal relevance and minimal redundancy criterion. We successfully test our method both in the contexts of image classification and microarray data classification. For most of the tested data sets, we obtain better classification results than those reported in the literature.

Keywords Filter feature selection · Mutual information · Entropic spanning graphs · Microarray

1 Introduction

Dimensionality reduction of the raw input variable space is a fundamental step in most pattern recognition tasks. Focusing on the most relevant information in a potentially overwhelming amount of data is useful for a better understanding of the data, for example in genomics [1, 2, 3]. A properly selected features set significantly improves classification performance. Thus, the removal of the noisy, irrelevant, and redundant features is a challenging task.

There are two major approaches to dimensionality reduction: feature selection and feature transform. Whilst feature selection reduces the feature set by discarding the features, which are not useful for some purpose (generally for classification), feature transform methods (also called feature extraction) build a new feature space from the original variables.

The literature differentiates among three kinds of feature selection: filter methods [4, 5], wrapper methods [6], and on-line [7]. Filter feature selection does not take into account, the properties of the classifier (it relies on statistical tests over the variables), while wrapper feature selection tests different feature sets by building the classifier. Finally, on-line feature selection incrementally adds or removes new features during the learning process.

Feature selection (FS) is a combinatorial computational complexity problem. FS methods must be oriented to find suboptimal solutions in a feasible number of iterations. On one hand, the algorithm which selects features from the set cannot be exhaustive, but it has to be suboptimal. On the other hand, the criterion evaluating the feature subsets is a delicate point. It has to estimate the usefulness of a subset accurately and inexpensively. The present work is neither centered on the selection algorithms, nor on stopping criteria, but on the feature selection criteria. When there are thousands of features, wrapper approaches become unfeasible because the evaluation of large feature sets is computationally expensive. Filter approaches evaluate feature subsets via different statistical measures.

B. Bonev (✉) · F. Escolano · M. Cazorla
Deptartamento Ciencia Computación e Inteligencia Artificial, Universidad de Alicante, Ap. Correos 99, 03080 Alicante, Spain
e-mail: boyan@dccia.ua.es

F. Escolano
e-mail: sco@dccia.ua.es

M. Cazorla
e-mail: miguel@dccia.ua.es

There are univariate and multivariate evaluation methods [8]. Among the univariate filter approaches, a fast way to evaluate individual features is given by their relevance to the classification, by maximizing the mutual information [9] between each variable and the classification output. As Guyon and Elisseeff state in [4], this is usually sub-optimal for building a predictor, particularly if the variables are redundant. Conversely, a subset of useful variables may exclude many redundant, but relevant, variables. To overcome this limitation, Peng et al. [10] minimize redundancy among the selected features set. Still a problem remains, as these criteria are based on individual comparisons between features. The reason for this is the fact that estimating mutual information (and entropy) in a continuous multi-dimensional feature space is a hard task.

In this work, we overcome the latter problem by using entropic spanning graphs to estimate mutual information [11, 12, 13]. The estimation's complexity does not depend on the number of dimensions, but on the number of samples. It allows us to estimate mutual information and thus, maximize the dependency between combinations of thousands of features and the class labels. We compare classification results to other MI-based filter feature selection criteria and perform experiments on gene patterns with thousands of features. There is also a work in which, Torkkola [14] uses non-parametric mutual information and quadratic divergence measure for learning discriminative feature transforms. However, their approach is for feature extraction, while the present work is centered on feature selection.

This paper is structured as follows: in "Estimation of mutual information and entropy", the estimation of mutual information ("Mutual information estimation") and entropy ("Entropy estimation with entropic spanning graphs") are detailed. Then in "Feature selection criteria", feature selection criteria are explained and complexity is discussed. Experimental results are presented in "Microarray data", which is divided in to "Image data" with image data classification and "Microarray data" with microarray data analysis. Finally, conclusions and future work are stated in "Conclusions and future work".

2 Estimation of mutual information and entropy

2.1 Mutual information estimation

Mutual information (MI) is used in filter feature selection as a measure of the dependency between a set of features **S** and the classification prototypes **C**. The final objective of feature selection is to minimize the classification error. An example of the relation between MI and classification error is shown in Fig. 1.

MI can be calculated in different ways. Neemuchwala et al. [15] studied the use of entropic graph for MI estimation. In our approach, we calculate MI based on entropy estimation:

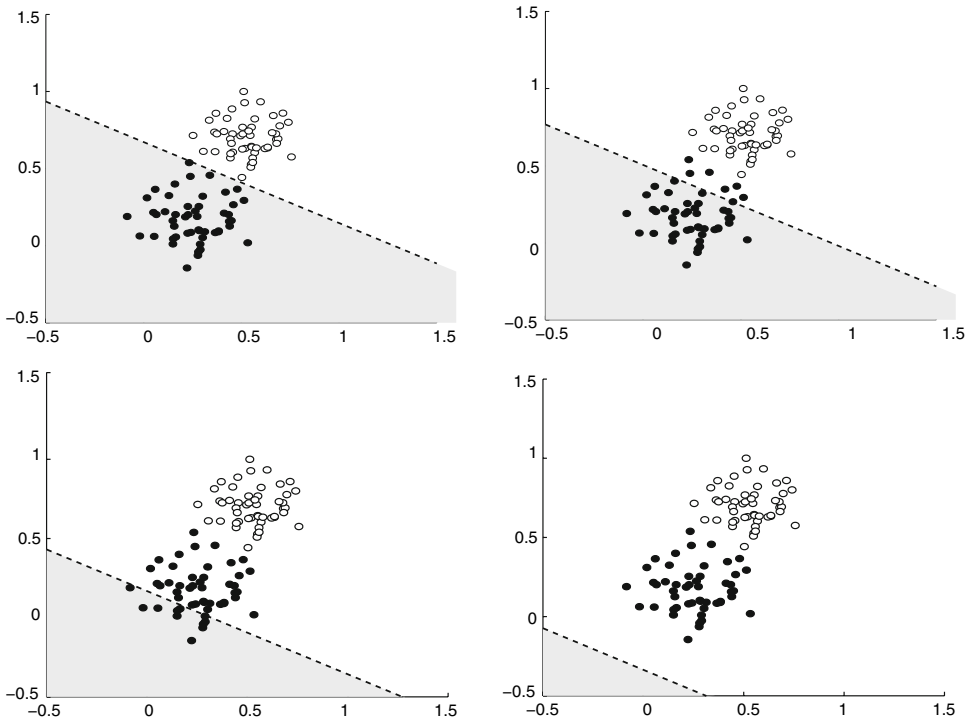

Fig. 1 Four different classifications of the same data, consisting of two Gaussian distributions. The mutual information between the data and their classes (obtained by means of entropic graphs) are from left to right and from top to bottom: 6.2067, 5.2767, 1.4046, and 0

$$I(\mathbf{S};\mathbf{C}) = \sum_{s\in\mathbf{S}}\sum_{c\in\mathbf{C}} p(s,c)\log\frac{p(s,c)}{p(s)p(c)} \quad (1)$$

$$= H(\mathbf{S}) - H(\mathbf{S}|\mathbf{C}) \quad (2)$$

$$= H(\mathbf{S}) + H(\mathbf{C}) - H(\mathbf{S},\mathbf{C}) \quad (3)$$

where s is a feature from the set of selected features $\mathbf{S}$ and c is a class label belonging to the set of prototypes $\mathbf{C}$.

Using the Eq. 2, the conditional entropy $H(\mathbf{S}|\mathbf{C})$ has to be calculated. To do this, $\sum H(X|C=c)p(C=c)$ entropies have to be calculated, and this is feasible insofar $\mathbf{C}$ is discrete ($\mathbf{C}$ consists of the class labelling). On the other hand, using Eq. 3, implies estimating the joint entropy. In our experiments, we used Eq. 2 because it is faster, due to the complexity of the entropy estimator, which depends on the number of samples as we will see in the following subsection.

2.2 Entropy estimation with entropic spanning graphs

Given a set of samples, we want to estimate the Shannon entropy of the probability density function of these samples [16, 17, 18]. The entropy estimation methods can be classified as "plug-in" and "non plug-in" methods. The "plug-in" methods consist in using the estimates of the density function in the expression of entropy [19]. One classical example is the Parzen window method. Its complexity is quadratic with respect to the number of dimensions. Another drawback of this estimator is its high variance and sensitivity to outliers.

Entropy estimation with entropic spanning graphs [11] is a "non plug-in" method because it estimates entropy directly from the set of samples. The main advantage is the possibility to work in a very high-dimensional space, in contrast to Parzen windows. With this approach, Shannon entropy can not be estimated directly. The first step is to estimate Rényi's α-entropy from the minimal spanning tree (MST) of the data (see Fig. 2). (K-nearest neighbors have also been used as α-entropy estimators [20], however, in our implementation, we use MST). The MST is the acyclic graph, spanning the data samples, with a minimal total edge length.

The set of features $\mathbf{S}$ contains continuous values, so each data sample $\boldsymbol{x}_i \in \mathbb{R}^d, d = |\mathbf{S}|$ has a mapping in a d-dimensional Euclidean space. The samples are points in the space and $\{e\}$ are the edges which connect them. Let $e_{ij} = (\boldsymbol{x}_i, \mathbf{x}_j)$ be the edges of the minimal spanning tree T, which connects all the data points and let the length of each edge $|e_{ij}|$ be the Euclidean distance $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|$. Given a power value $\gamma \in (0, d)$, the weighted length of the MST is defined as follows:

$$L_\gamma(\{\boldsymbol{x}_i\}) = \min_{\{e\}\in T} \sum_{e_{ij}\in\{e\}} |\, e_{ij}\,|^\gamma \quad (4)$$

where $\{\boldsymbol{x}_i\}$ are the data samples, in the feature space $\mathbf{S}$.

There are different algorithms for building the MST of a set of samples. Tarjan's algorithm [21] has a linear expected complexity $O(m)$ with respect to the number of edges m. In practice, it is convinient to use other algorithms with a similar complexity and simpler implementation. Such is the algorithm presented in [22], for which the complexity is $O(m + n \log n)$, being n, the number of vertices, however, when $m \gg n$, the complexity is nearly linear. Moreover, the constant factors are computationally less expensive than those of the Tarjan's algorithm.

Rényi's α-entropy can be directly estimated from the weighted length definition presented in Eq. 4. The α-entropy of the probability density function p is defined as:

$$H_\alpha(p) = \frac{1}{1-\alpha}\ln\int p^\alpha(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, 0 \le \alpha < 1 \quad (5)$$

being

$$\lim_{\alpha\to 1} H_\alpha(p) = -\int p(\boldsymbol{x})\ln p(\boldsymbol{x})\mathrm{d}\mathbf{x} \quad (6)$$

which means that when $\alpha \to 1$, the α-entropy converges to the Shannon entropy. Consequently, in a feature space with more than two dimensions $d \ge 2$, the following estimator is asymptotically stable and consistent, as showed in [23]:

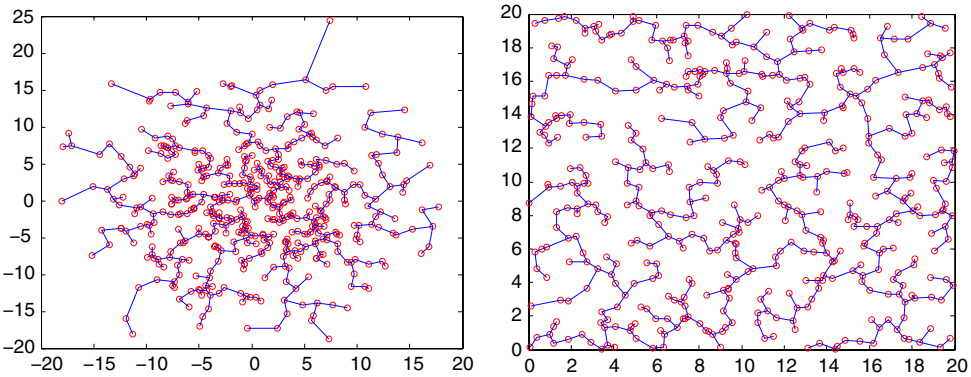

Fig. 2 The minimal spanning trees of bi-dimensional data. The MST of the Gaussian distribution (on the *left*) is shorter than the MST of the non-Gaussian distribution (on the *right*)

$$H_\alpha(\{\boldsymbol{x}_i\}) = \frac{d}{\gamma}\left[\ln\frac{L_\gamma(\{\boldsymbol{x}_i\})}{n^\alpha} - \ln\beta_{L_\gamma,d}\right] \tag{7}$$

where

$$\alpha = (d-\gamma)/d \tag{8}$$

and $\beta_{L_\gamma,d}$ is a constant not depending on p. An approximation [24] that can be used for large d is:

$$\beta_{L_\gamma,d} \approx \frac{\gamma}{2}\ln\frac{d}{2\pi e} \tag{9}$$

We use the Eq. 7 for estimating α-entropy. In the expression of the α-entropy (Eq. 5), it can be seen that $\alpha \neq 1$, otherwise there would be a division by zero. Therefore, Shannon entropy can not be directly estimated. The solution is to approximate the value of H_α for $\alpha = 1$ by means of a continuous function that captures the tendency of H_α in the environment of 1. This function is obtained by changing the weight exponent γ. This enables us to estimate $H_\alpha(p)$ for different values of α (see Eq. 8) without having to recompute the MST. That function is monotonous decreasing, and allows us to find the α^* value appropriate for extrapolating the correct entropy value. In [25], it is experimentally verified that α^* is constant for a fixed number of samples and dimensions, and for different covariance matrices. In their work, they use entropic spanning graphs as entropy estimators for solving a clustering problem. In our work, we exploit them for solving a feature selection problem by means of mutual information estimation.

3 Feature selection criteria

There are different filter feature selection criteria for selecting or discarding a feature or a feature set. Peng et al. [10] studied the possibility to maximize the dependency between the feature set **S** and the prototypes **C** (max-dependency criterion):

$$\max_{\mathbf{S}\subseteq\mathbf{F}} I(\mathbf{S};\mathbf{C}) \tag{10}$$

They find it unfeasible because the entropy estimation in high-dimensional feature spaces (**F**) is very hard, and yields poor results, due to the entropy estimator they use. Instead, they maximize the relevance $I(x_j;\mathbf{C})$ of each individual feature $x_j \in \mathbf{F}$ and at the same time, minimize the redundancy between x_j and the rest of selected features $x_i \in S$, $i \neq j$. This is the max-relevance min-redundance (mRMR) criterion and its formulation for the selection of the mth feature is:

$$\max_{x_j\in\mathbf{F}-\mathbf{S}_{m-1}}\left[I(x_j;\mathbf{C}) - \frac{1}{m-1}\sum_{x_i\in\mathbf{S}_{m-1}} I(x_j;x_i)\right] \tag{11}$$

In this work, we state that the max-dependency criterion is feasible, even for very high-dimensional feature spaces. The complexity of MI estimation, explained in "Estimation of mutual information and entropy", depends on the entropic spanning graph construction, which has a computational complexity of $O[s\log(s)]$ order, where s is the number of samples. The number of dimensions increases computational time linearly, so it is not bounded by the curse of dimensionality. Also, the accuracy of the estimation does not depend on the dimensionality. The MD criterion evolution and results are illustrated further, on "Microarray data" (Fig. 9).

We also propose the max-min-dependency (MmD) criterion (Eq. 12), which adds to the max-dependency (MD) criterion, the minimization of the mutual information between the set of discarded or not selected features and the classes:

$$\max_{\mathbf{S}\subseteq\mathbf{F}}[I(\mathbf{S};\mathbf{C}) - I(\mathbf{F}-\mathbf{S};\mathbf{C})] \tag{12}$$

The aim of the MmD criterion is to avoid leaving out features, which have information about the prototypes. In the Fig. 3, we can see the evolution of the criterion and the terms $I(\mathbf{S};\mathbf{C})$ and $I(\mathbf{F}-\mathbf{S};\mathbf{C})$, together with the tenfold CV and test errors, which the FS yielded. The experiment is performed on image data, which will be explained in detail in "Image data".

A different issue is the way that feature combinations are generated. An exhaustive search among the features set combinations would present a $O(n!)$ combinatorial complexity, where n is the total number of features. For our experiments, we have used a greedy forward feature selection algorithm, which starts from a small feature set, and adds one feature at a time.

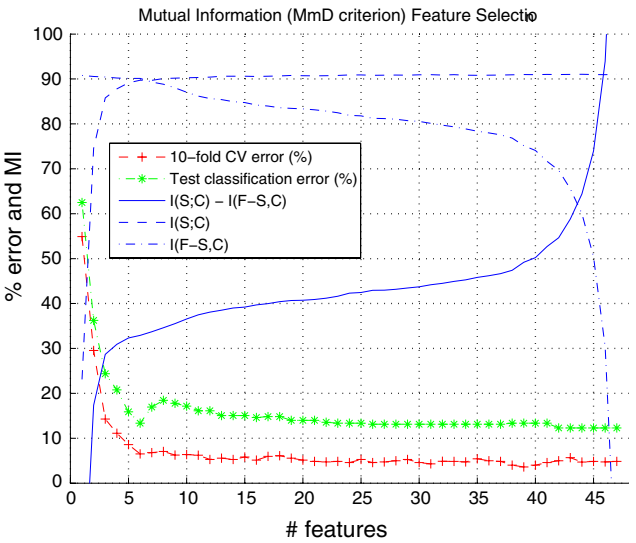

Fig. 3 Maximum–minimum dependency feature selection criterion on image data with 48 features

Therefore, with the mRMR criterion, each iteration would consists of calculating the MI between a feature and the prototypes, as well as the MI between that feature and each one of the already selected ones (see Eq. 11). Such search performs $\sum_{i=1}^{n} i(n-i+1)$ estimations of the MI, which has a $O(n^3+n^2+n)$ computational complexity. Using the MD criterion instead, requires just one MI calculus per iteration. The total number of MI estimations is $\sum_{i=1}^{n} n-i+1$, which has a $O(n^2+n)$ computational complexity. Using the MmD criterion increases twice the computational time, although the complexity remains of the same order as for MD.

4 Experiments

The aim of these experiments is to show the performance of the FS criteria on high-dimensional data classification problems found in real pattern recognition applications. The first experiment is on image data and we compare the mRMR criterion with the MD and MmD criteria. The other experiments are applications to microarray gene selection. For them, we only tested the MD and MmD criteria, as the mRMR is too complex for so many dimensions (more than 2,000).

4.1 Image data

The image data experiment consists of classifying the images taken from a human point of view, around rooms, corridors, stairs, and an outdoor route. Its application is environment recognition, as well as finding the most similar image in the training set. The approach is an appearance-based one, based on applying a large set of low-level filters (like corner detectors, edge detectors, and color filters) to the images, for applying feature selection to their histograms. Biological evidences [26, 27] report that low-level filters play an important role in biological visual recognition systems. For example, Gabor filters model the visual processing carried out by the simple and complex cells of the primary visual cortex of higher mammals. The organization of these cells results from an unsupervised learning in the visual system, during the first months of life [28]. In computer vision for object recognition, Gabor filters, Haar features, steerable filters [29], and color cooccurrence histograms [30, [31] are being widely used, providing tolerance to noise and robustness to small perspective changes and occlusions. On the other hand, this approach is appropriate for real time classification, as the FS is an offline process. Then, applying the selected filters to a test image and classifying it, can take about 0.1 s on a standard PC, depending on the number of filters and the classifier.

In our experiment, the data comes from a set of 721 (training set) + 470 (test set) images (samples) with a 320 × 240 resolution, labeled with six different classes. A set of 16 different low-level filters is applied to the images. Features consist of single bins of the filters' histograms. Among the filters, there are some color filters, corner and edge detectors, and range information obtained from a stereo camera. An example is shown on Fig. 4.

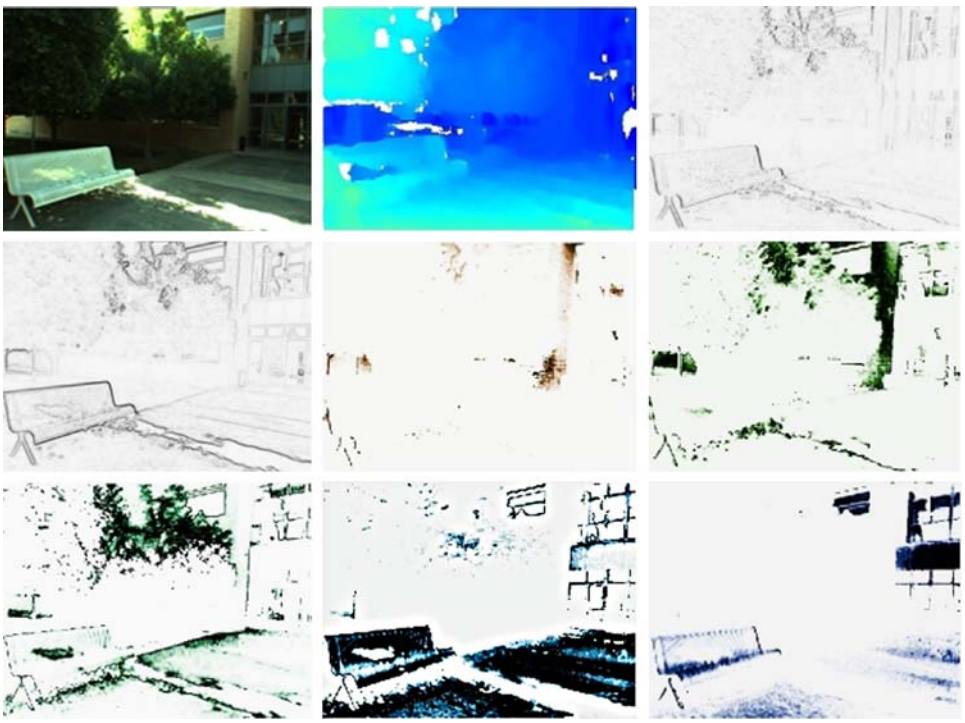

Fig. 4 Some selected filters. From top–bottom and left–right: input image I, depth Z, vertical gradient ∇_y, gradient magnitude $|\nabla|$, and the color filters: H_1–H_4. Bins from filters H_8–H_{10} were also selected but are not showed because they yield null output for this input image

The total number of features extracted from the images can be varied by changing the number of bins of the filter histograms. We have experimentally found that for the present experiment, a good number of bins is four. In Fig. 6, it can be seen that two-bins generate such a small number of features, that the tenfold cross validation (CV) error remains too high. On the other hand, more than four bins, while increasing the total number of feature unnecessarily, do not minimize the error very much. The classification performance also depends on the number of classes in which, the data set is divided. In Fig. 6, we show that a lower number of classes yield better classification performances. For the present experiment, we have divided the images in six different environments: an office, two corridors, stairs, entrance, and a tree avenue.

After the filter feature selection process, the feature sets have to be tested with a classifier in order to see the classification performances they yield. We use a K-nearest neighbour (K-NN, with $K = 1$) classifier because we need to know both the class and the neighbours of a given image. An example with test images and their nearest neighbours from the training set, is shown on Fig. 7. We have also compared the performances of the K-NN and a support vector machine (SVM) classifier and the latter does not outperform K-NN for the present classification problem. In Table 1, we show the confusion matrices of six-class classifications for K-NN and SVM.

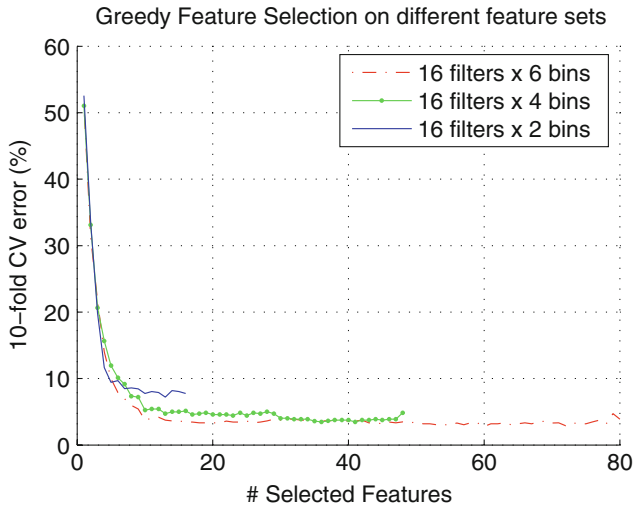

Fig. 5 Finding the optimal number of bins N_b

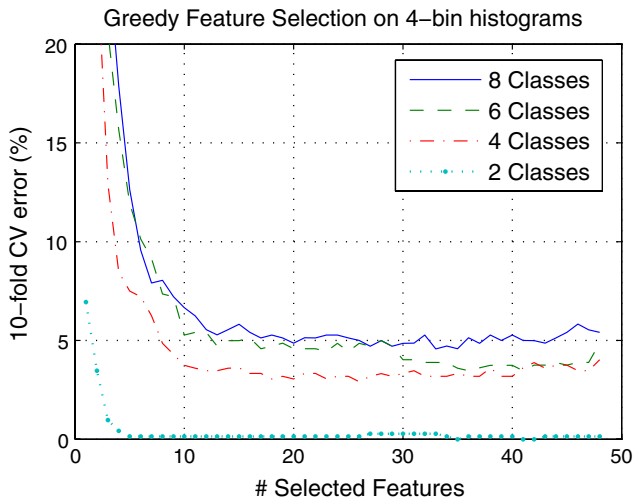

Fig. 6 Evolution of the CV error for different number of classes N_c

In Fig. 8, we show the classification errors of the feature sets selected with the MD, MmD and mRMR criteria. Only 20 selected features are represented, as for larger features sets, the error does not decrease. The tenfold cross validation error of K-NN is represented, as well as the test-error, which was calculated using the additional test set of images, containing 470 samples.

With mRMR, the lowest CV error (8.05%) is achieved with a set of 17 features, while with MD a CV error of 4.16% is achieved with a set of 13 features. The CV errors yielded by MmD are very similar to those of MD. On the other hand, test errors present a sligthly different evolution. The MmD test error descends faster than the MD test error, only for the first seven feature sets. For the rest of the feature sets, the best error is given by MD, and the worst one is given by mRMR. In report by Peng et al. [10], the experiments yielded better results with the mRMR criterion than with the MD criterion. Contrarily, we obtain better performance using MD. This may be explained by the fact that the entropy estimator we use does not degrade its accuracy as the number of dimensions increases.

4.2 Microarray data

In order to illustrate the dimensionality independence of the entropy estimator we use, we present classification experiments on some well-known microarray data sets. An application of feature selection to microarray data is to identify small sets of genes with good predictive performance for diagnostic purposes [32]. Traditional gene selection methods often select genes according to their individual discriminative power. Such approaches are efficient for high-dimensional data but cannot discover redundancy and basic interactions among genes. The contribution of the present work is the efficient evaluation of whole sets of features.

First, we will evaluate the MD and the MmD criteria with experiments on the NCI data set and next, we will show results for other microarray data sets. The NCI data set contains 60 samples (patients), each one containing 6,380 dimensions (genes). The samples are labeled with 14 different classes of human tumor diseases. The purpose of feature selection is to select those genes which are useful for predicting the disease.

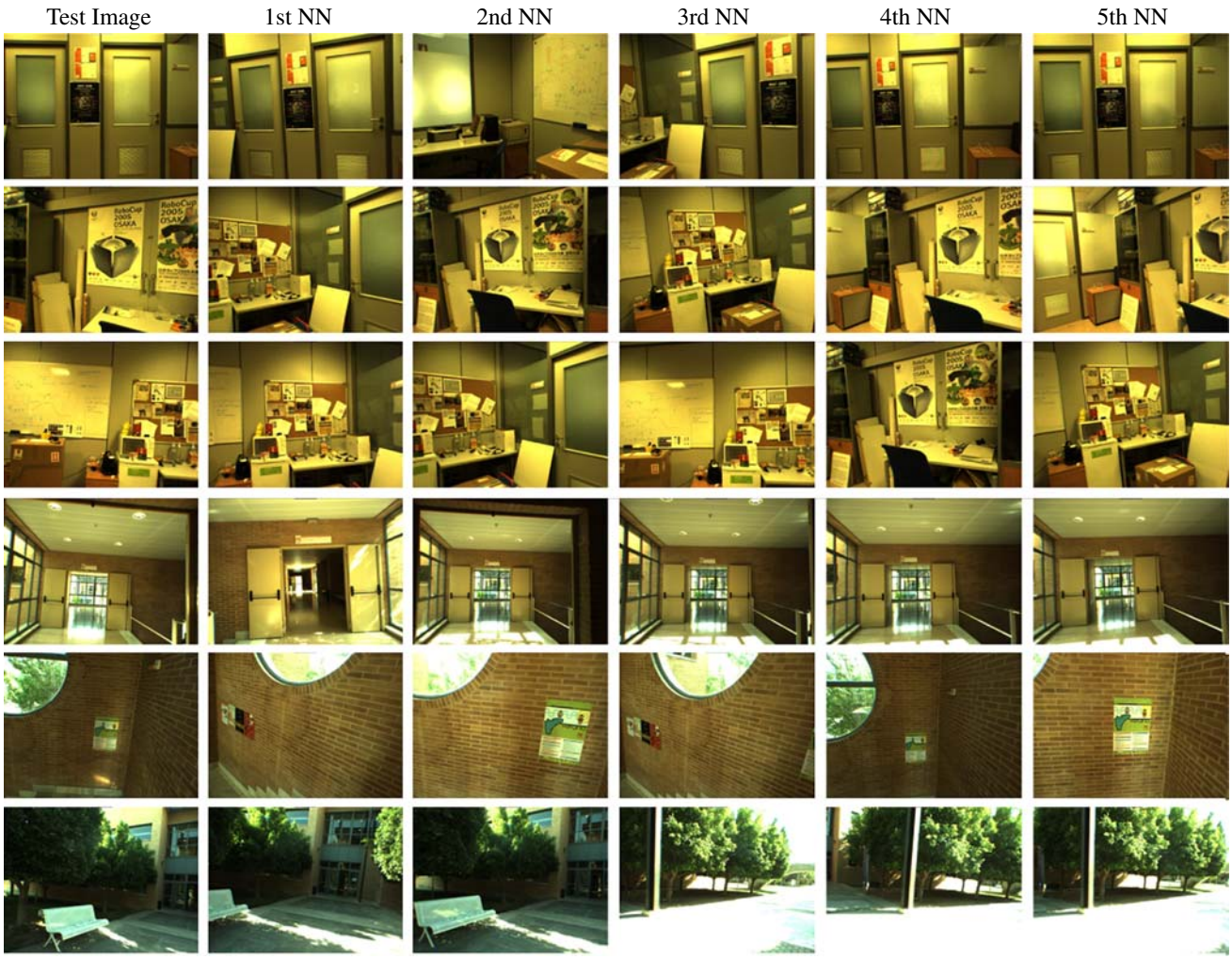

Fig. 7 The nearest neighbours of different test images. The training set from which the neighbours are extracted contains 721 images taken during an indoor–outdoor walk. The amount of low-level filters selected for building the classifier is 13 out of 48 in total

Table 1 *K*-NN/SVM confusion matrix

	C#1	C#2	C#3	C#4	C#5	C#6
C#1	26	0	0	0	0	0
C#2	2/3	63/56	1/4	0/3	0	0
C#3	0	1/0	74/67	1/9	0	0
C#4	4/12	5/6	10/0	96/95	0/2	0
C#5	0	0	0	0	81	0
C#6	0	0	0	0	30/23	78/85

In Fig. 9, we have represented the increase of mutual information, and the resulting leave one out cross validation error[1] (LOOCV). The error decreases until 39 features are selected, and then it slowly increases, due to the addition of redundant and noisy genes.

We also tested the MmD criterion on the microarray data sets, and the resulting performances were similar (see Table 2), being slightly better, the error rates yielded by MD. In Fig. 10, we can see that for the NCI data set, the first 16 selected features are the same for both criteria. For the subsequent feature sets, firstly the MmD criterion achieves lower error, but with the addition of more features, MD wins. In order to better understand the behaviour of both criteria we have represented the gene expression matrix of the best feature sets selected by MD and MmD in the Fig. 11. There are 24 genes selected by both criteria, and despite the rest of the genes are different, the overall expression matrices keep a similar aspect. This means that there are different genes with similar expression, and therefore with similar information about the prototypes.

In Fig. 9, we show the leave one out cross validation errors for the selected feature subsets, using the max-

[1] LOOCV measure is used when the number of samples is so small that a test set cannot be built. It consists of building all possible classifiers, each time leaving out only one sample for test. Note that in this work, it is used just for evaluating the results, but not as a selection criterion.

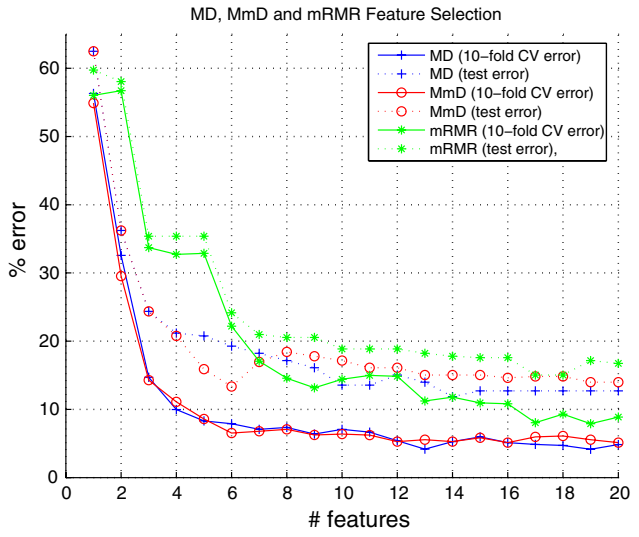

Fig. 8 Feature selection performance on image histograms data with 48 features. Comparison between the maximum dependency (MD), maximum–minimum-dependency (MmD) and the minimum-redundancy maximum-relevance (mRMR) criteria

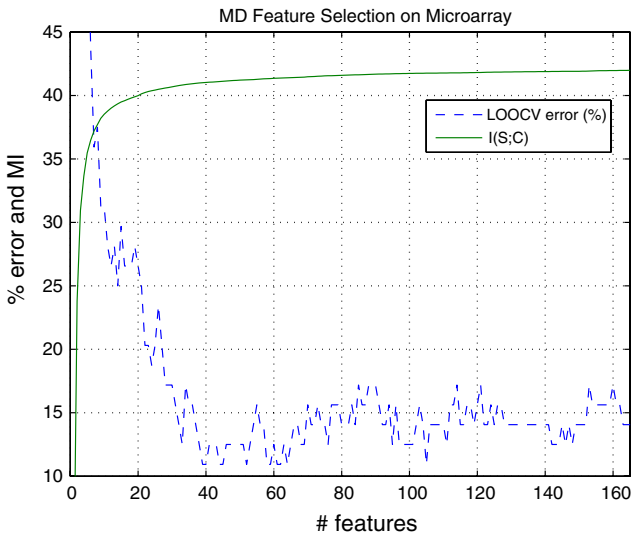

Fig. 9 Maximum dependency feature selection performance on the NCI microarray data set with 6,380 features. The mutual informacion of the selected features is represented. The FFS algorithm using the maximum dependency criterion obtained the lowest LOOCV error with a set of 39 features

dependency criterion. Only the best 220 genes (out of 6,380) are on the x-axis, and it took about 24 h on a PC with Matlab to select them. During this time MI was calculated $\sum_{i=1}^{220}(6{,}380 - i + 1) = 1{,}385{,}670$ times.

In [33], an evolutionary algorithm is used for feature selection and the best LOOCV error achieved is 23.77% with a set of 30 selected features. In our experiment, we achieve a 10.94% error with 39 selected features. In addition to the experiments with the NCI data set, we have performed FS experiments on other four well-known microarray expression data sets:[2]

- Leukemia: the training data set consists of 38 bone marrow samples labelled with two classes, 27 acute lymphoblastic leukemia (ALL) and 11 acute myeloid leukemia (AML), over 7,129 probes (features) from 6,817 human genes. Also 34 samples testing data is provided, with 20 ALL and 14 AML. We obtained a 2.94% test error with seven features, while [34] reports the same error selecting 49 features via individualized markers. Other recent works [35, 3] report test errors higher than 5% for the leukemia data set.
- Colon: this data set contains 62 samples collected from colon-cancer patients. Among them, 40 tumor biopsies are from tumors and 22 normal biopsies are from healthy parts of the colons of the same patients. From 6,500 genes, 2,000 were selected for this data set, based on the confidence in the measured expression levels. Our feature selection process selects 15 of them, resulting in a 0% LOOCV error, while recent works report errors higher than 12% [35, 36].
- Central nervous system embryonal tumors: this data set contains 60 patient samples, 21 are survivors (patients who are alive after treatment) and 39 are failures (those who succumbed to their disease). There are 7,129 genes in the data set. We selected nine genes achieving a 1.67% LOOCV error. In [34], a 8.33% CV error with 100 features is reported.
- Prostate: this data set contains two classes for tumor versus normal classification. The training set contains 52 prostate tumor samples and 50 non-tumor prostate samples with 12,600 genes. An independent set of testing samples is also available, but it is from a different experiment and has a nearly tenfold difference in overall microarray intensity from the training data, so we have applied a 0.1 correction for our experiments. The test set is the one used in [37], where extra genes contained in the testing samples are removed, and there are 25 tumor and 9 normal samples. Our experiments on the prostate data set achieved the best error with only five features, with a test error of 5.88%. Other works [35, 3] report test errors higher than 6%.

Only Gentile [3] reports a little better classification error for the leukemia (2.50% with 100 features) data set, but we cannot compare it to our results because their results refer to different training/test splits. The best feature selection results achieved with our method are summarized in the Table 2.

[2] Datasets can be downloaded from the Broad Institute http://www.broad.mit.edu/, Stanford Genomic Resources http://genome-www.stanford.edu/, and Princeton University http://microarray.princeton.edu/.

Table 2 Feature selection results on different microarray data sets

Criterion	Errors (%)			Selected features	
	LOOCV	10FCV	Test	$\|S\|$	S
Colon data set, $\|F\| = 2{,}000$					
MD/MmD	0.00	0.00	N/A	15	1 87 295 371 534 625 653 698 858 1024 1042 1058 1161 1212 1423
NCI data set, $\|F\| = 6{,}830$					
MD	10.94	10.48	N/A	39	19 135 246 663 692 766 982 1177 1470 1671 2080 3227 3400 3964 4057 4063 4110 4289 4357 4441 4663 4813 5226 5481 5494 5495 5508 5752 5790 5892 6013 6019 6032 6045 6087 6145 6184 6286 6643
MmD	12.50	13.81	N/A	35	15 19 246 294 663 717 982 1728 2080 2112 3400 3964 4057 4063 4110 4289 4357 4441 4663 4813 5481 5494 5495 5508 5853 5871 5892 6032 6045 6087 6145 6263 6310 6312 6459
Leukemia (ALL-AML) data set, $\|F\| = 7{,}129$					
MD	0.00	0.00	2.94	7	1779 2015 2288 4847 6182 6277 7093
MmD	0.00	0.00	5.88	6	1779 2015 4847 6182 6277 7093[a]
Central Nervous System, $\|F\| = 7{,}129$ data set					
MD	1.67	0.00	N/A	15	454 491 560 863 1506 1667 1777 1879 2474 2548 4994 6063 6143 6165
MmD	1.67	1.67	N/A	15	6634
Prostate data set, $\|F\| = 12{,}600$					
MD	0.00	0.91	5.88[b]	14	289 1770 3160 3824 4567 6185 6359 6443 6462 7756 8351 9172 9332 10234

Test, leave one out cross validation (LOOCV), and tenfold cross validation (10FCV) errors are provided. For data sets which do not provide a test set, such error is annotated as not available (N/A). S the selected features subset, $|S|$ the number of selected features, $|F|$ the total number of features in the data set. Max-dependency (MD) and max–min-dependency (MmD) criteria are compared. The experiments where both criteria yield the same results, are written in one row (MD/MmD)

[a] The first six features selected are the same as for MD, however, the next feature selected is different (the 2402), and the test error increases to 8.82%

[b] Such test error is also achieved with only five features (160 6185 7756 9172 10234), but the LOOCV error they produce is higher (2.91% LOOCV)

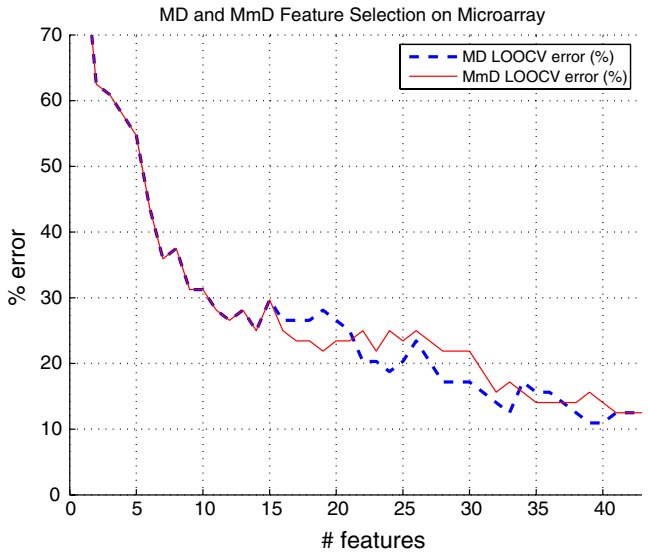

Fig. 10 Maximum dependency and maximum–minimum dependency feature selection performance on the NCI microarray. Only the first 43 features are represented, out of 6,380

5 Conclusions and future work

In this paper, we have presented a filter feature selection approach based on mutual information. The mutual information estimation depends linearly on the number of features, and it depends n-logarithmically on the number of samples. Therefore this approach circumvents the curse of dimensionality for high-dimensional patterns, such as DNA microarray data. In contrast to wrapper approaches, this filter approach does not rely on minimizing the classification error, but on maximizing MI of sets of features and the prototypes. However as a consequence of this, the classification error actually decreases.

We obtain better results by evaluating the whole feature subsets with the max-dependency criterion, than comparing features one by one with the min-redundancy max-relevance criterion. Also, the max–min-dependency criterion does not yield better error rates than max-dependency. Classification experiments were performed on image data and five well-known microarray data sets. For most of them, we obtained better classification results than those reported in the

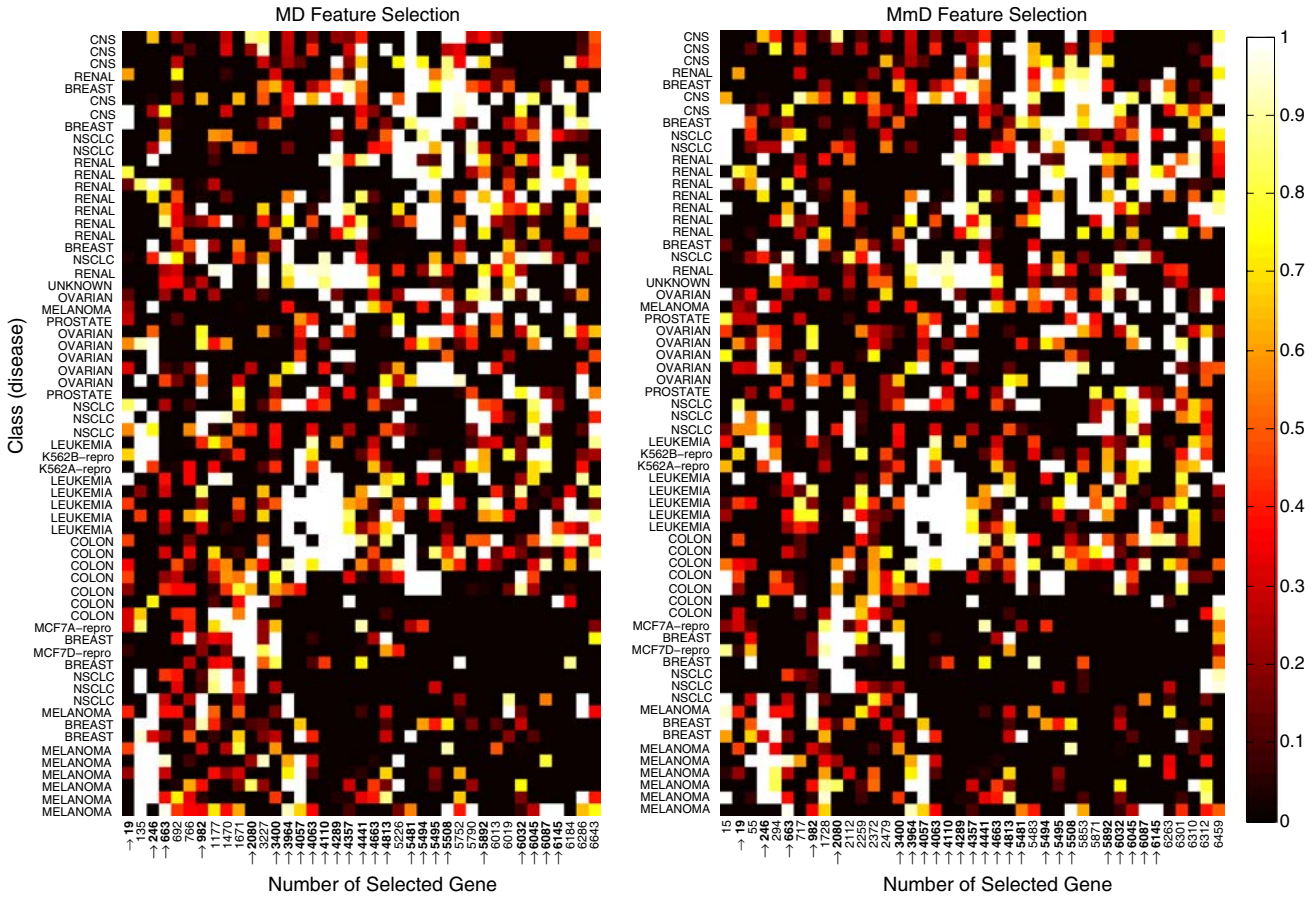

Fig. 11 Feature selection on the NCI DNA microarray data. The MD (on the *left*) and MmD (on the *right*) criteria were used. Features (genes) selected by both criteria are marked with an *arrow*

literature. Also, we achieve the highest classification performance with smaller feature sets than those obtained with the state of the art feature selection criteria.

In the future, we want to explore feature selection algorithms different than FFS. This algorithm starts selecting small feature subsets, but with our approach, it would not be hard to start evaluating larger feature subsets and removing the less informative ones.

Acknowledgments This research is funded by the project DPI2005-01280 from the Spanish Government.

References

1. Sima C, Dougherty ER (2006) What should be expected from feature selection in small-sample settings. Bioinformatics 22(19):2430–2436
2. Xing EP, Jordan MI, Karp RM (2001) Feature selection for high-dimensional genomic microarray data. In: Proceedings of the 18th international conference on machine learning 601–608
3. Gentile C (2003) Fast feature selection from microarray expression data via multiplicative large margin algorithms. In: Thrun S, Saul L, Schölkopf B (eds) Advances in Neural Information Processing Systems 16. MIT Press, Cambridge
4. Guyon I, Elisseeff A (2003) An introduction to variable and feature selection. J Mach Learn Res 3:1157–1182
5. Abe N, Kude M, Toyama J, Shimbo M (2006) Classifier-independent feature selection on the basis of divergence criterion. Pattern Anal Appl 9(2):127–137
6. Blum AL, Langley P (1997) Selection of relevant features and examples in machine learning. Artif Intell 97(1–2):245–271
7. Perkins S, Theiler J (2003) Online feature selection using grafting. In: Proceedings of the 20th international conference on machine learning (ICML-2003), Washington
8. Harol A, Lai C, Pekalska E, Duin RPW (2007) Pairwise feature evaluation for constructing reduced representations. Pattern Anal Appl 10(1):55–68
9. Cover T, Thomas J (1991) Elements of information theory. Wiley, New York
10. Peng H, Long F, Ding C (2005) Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans Pattern Anal Mach Intell 27(8):1226–1238
11. Hero AO, Michel O (2002) Applications of entropic spanning graphs. IEEE Signal Process Mag 19(5):85–95
12. Zyczkowski K (2003) Renyi extrapolation of Shannon entropy. Open Syst Inf Dyn 10(3):298–310
13. Mokkadem A (1989) Estimation of the entropy and information of absolutely continuous random variables. IEEE Trans Inf Theory 35(1):193–196
14. Torkkola K Feature (2003) Extraction by non-parametric mutual information maximization. J Mach Learn Res 3:1415–1438

15. Neemuchwala H, Hero A, Carson P (2006) Image registration methods in high-dimensional space. Int J Imaging Syst Technol 16(5):130–145
16. Paninski I (2003) Estimation of entropy and mutual information. Neural Comput 15(1):
17. Wolpert D, Wolf D (1995) Estimating function of probability distribution from a finite set of samples. Los Alamos National Laboratory Report LA-UR-92-4369, Santa Fe Institute Report TR-93-07-046
18. Wachowiak P, Smolíková R, Tourassi D, Elmaghraby S (2005) Estimation of generalized entropies with sample spacing. Pattern Anal Appl 8(1–2):95–101
19. Beirlant E, Dudewicz E, Gyorfi L, Van der Meulen E (1996) Nonparametric entropy estimation. Int J Math Stat Sci 5(1):17–39
20. Oubel E, Neemuchwala H, Hero A, Boisrobert L, Laclaustra M, Frangi AF (2005) Assessment of artery dilation by using image registration based on spatial features. In: Proceedings of SPIE medical imaging, April 2005, vol 5747, pp 1283–1291
21. Karger DR, Klein PN, Tarjan RE (1995) A randomized linear-time algorithm to find minimum spanning trees. J ACM 42(2): 321–328
22. Katriel I, Sanders P, Träff J (2003) A practical minimum spanning tree algorithm using the cycle property. 11th European Symposium on Algorithms(ESA), LNCS No. 2832, 679–690
23. Hero AO, Michel O (1999) Asymptotic theory of greedy aproximations to minnimal k-point random graphs. IEEE Trans Inf Theory 45(6):1921–1939
24. Bertsimas DJ, Van Ryzin G (1990) An asymptotic determination of the minimum spanning tree and minimum matching constants in geometrical probability. Oper Res Lett 9(1):223–231
25. Peñalver A, Escolano F, Sáez JM (2006) EBEM an entropy-based EM algorithm for Gaussian mixture models. ICPR 451–455
26. Tarr MJ, Bülthoff HH (1999) Object recognition in man, monkey, and machine. Cognition Special Issues, MIT Press, Massachusetts
27. Dill M, Wolf R, Heisenberg M (1993) Visual pattern recognition in Drosophila involves retinotopic matching. Nature 365(6448):639–644
28. Meese TS, Hess RF (2004) Low spatial frequencies are suppressively masked across spatial scale, orientation, field position, and eye of origin. J Vis 4(10):843–859
29. Carmichael O, Mahamud S, Hebert M (2002) Discriminant filters for object recognition. Technical report, Robotics Institute, Carnegie Mellon University, March, CMU-RI-TR-02-09
30. Ekvall S, Kragic D, Hoffmann F (2005) Object recognition and pose estimation using color cooccurrence histograms and geometric modeling. Image Vis Comput 23:943–955
31. Chang P, Krumm J (1999) Object recognition with color cooccurrence histograms. In: IEEE conference computer vision pattern recognition, Fort Collins, June 23–25
32. Stolovitzky G (2003) Gene selection in microarray data: the elephant, the blind men and our algorithms. Curr Opin Struct Biol 13(3):370–376
33. Jirapech-Umpai T, Aitken S (2005) Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes. BMC Bioinformatics 6:148
34. Pavlidis P, Poirazi P (2006) Individualized markers optimize class prediction of microarray data. BMC Bioinformatics 7:345
35. Díaz-Uriate R, Alvarez de Andrés S (2006) Gene selection and classification of microarray data using random forest. BMC Bioinformatics 7(1):3. doi:10.1186/1471-2105-7-3
36. Ruiz R, Riquelme JC, Aguilar-Ruiz JS (2006) Incremental wrapper-based gene selection from microarray data for cancer classification. Pattern Recognit 39(12):2383–2392
37. Singh D, Febbo PG et al. (2002) Gene expression correlates of clinical prostate cancer behavior. Cancer Cell 1(2):203–209. doi: 10.1016/s1535-6108(02)00030-2

Author Biographies

Boyan Bonev received a BS Degree in Computer Science from the University of Alicante (Spain) in 2005. He is currently Research Assistant with the Department of Computer Science and Artificial Intelligence of the University of Alicante. His research interests areas are pattern recognition, computer vision and mobile robotics. He has participated in the international RoboCup competition with the spanish team. He is currently working on his Ph.D. thesis focused on feature selection for pattern recognition.

Francisco Escolano received his Bachelors degree in Computer Science from the Polytechnical University of Valencia (Spain) in 1992 and his Ph degree in Computer Science from the University of Alicante in 1997. Since 1998, he is an Associate Professor with the Department of Computer Science and Artificial Intelligence of the University of Alicante. He has been post-doctoral fellow with Dr. Norberto M. Grzywacz at the Biomedical Engineering Department of the University of South California in Los Angeles, and he has also collaborated with Dr. Alan L. Yuille at the Smith-Kettlewell Eye Research Institute of San Francisco. Recently, he visited the Liisa Holm's Bioinformatics Lab at the University of Helsinki. His research interests are focused on the development of efficient and reliable computer-vision algorithms for biomedical applications, active vision and video-based surveillance. He is also interested in the coupling between computer and biological vision. He is the head of the Robot Vision Group.

Miguel Cazorla received a BS degree in Computer Science from the University of Alicante (Spain) in 1995 and a PhD in Computer Science from the same University in 2000. He is currently Associate Professor with the Department of Computer Science and Artificial Intelligence of the University of Alicante. He has been post-doctoral fellow with Dr. Alan L. Yuille at the Smith-Kettlewell Eye Research Institute of San Francisco and with Dr. Sebastian Thrun at Robot Learning Lab at Carnegie-Mellon University. His research interest areas are computer vision and mobile robotics (mainly using vision to implement robotics tasks)