KERNEL-BASED CLASSIFICATION OF TISSUES USING FEATURE WEIGHTINGS

A. KERTÉSZ-FARKAS* – A. KOCSOR

Research Group on Artificial Intelligence of the
Hungarian Academy of Sciences and University of Szeged
H-6720 Szeged, Aradi vértanúk tere 1., Hungary
(phone: +36-62-544-140; fax: +36-62-425-508)

**e-mail: kfa@inf.u-szeged.hu*

(Received 10th Sep 2005, accepted 10th Oct 2006)

Abstract. In high-dimensional spaces classification methods could be more effective using various feature selection methods. The training procedure could be speeded up by decreasing the dimension of the feature space, and the classification method could be improved by removing noisy or irrelevant features. In this paper we present a new method which weights the features according to their importance instead of removing the negligible ones via kernel functions. It could be applied to a range of real-world problems. We tested it on several biological datasets like a small part of the UCI Learning Repository and SCOP and the Leukaemia AML-ALL databases, and obtained a significantly better classification performance than that using the usual unweighted method.
Keywords: *Support Vector Machines (SVMs), classification, kernel functions, feature ranking algorithms*

Introduction

Biological databases like those generated by a DNA microarray consist of some thousands of features (i.e. components) that are not equally important. During classification, some features may be considered crucial while others can be safely ignored. It is obvious that the various features should have different weights in the classification procedure, i.e. the features should be weighted according to their importance. One such method is the Fisher Correlation Coefficients which assigns the following weighting value to the *i*th feature:

$$\frac{(\mu_i^+ - \mu_i^-)^2}{(\sigma_i^+)^2 + (\sigma_i^-)^2}$$

where μ_i^+, μ_i^-, σ_i^+, σ_i^- are the mean and standard deviations of the *i*th feature value for the positive and negative examples, resepectively [4].

In this paper we give a feature weighting method that is based on feature ranking. To rank the features here several methods are used and the corresponding weight calculation is based on this rank. A sample vector which is weighted by the given weights is computed via a kernel function. We carried out several experiments on a DNA microarray database and a part of the Astral databases of the SCOP protein sequence databases. We employed Support Vector Machines (SVM) with kernel functions as the classification method to obtain the experimental results.

The article is organised as follows. In Section 2 we discuss SVMs, feature weightings via kernels. In Section 3 we provide a summary of widely-used feature ranking methods. Section 4 describes the results of our method on real-world databases then, in Section 5, we discuss these results and their implications.

http://www.ecology.uni-corvinus.hu ● ISSN 1589 1623

SVM and kernels

SVM is a supervised binary classifier first introduced by Vapnik et al [1]. Let $D = \{(x, y) : x \in \Re^n, y \in \{-1,1\}\}$, where x is a n-dimensional training vector and y is the class label of x. $\Re^n$ is called the input space and D is referred to as the training database. Training an SVM amounts to solving an optimisation problem that determines a linear classification rule $f(x) = \langle w, x \rangle + b$. A test example z is classified as positive (or negative) if $f(z) > 0$ (or $f(z) < 0$). Such a classification rule determines a linear hyperplane decision boundary with normal vector w and bias term b that separates positive and negative classes.

The key part of SVM is the inner product $\langle x_i, x_j \rangle$ of two vectors x_i, x_j over $\Re^n$, which is used for the classification of samples. Given a feature map Φ from an input space to a (possibly infinite dimensional) dot product space (referred to as the kernel feature space), we obtain a inner product $\langle \Phi(x), \Phi(y) \rangle$. If a function $\kappa(x, y)$ is symmetric, continuous and positive definite, which is called the kernel function, then exist a Φ mapping so that $\kappa(x, y) = \langle \Phi(x), \Phi(y) \rangle$. We can directly and efficiently compute the kernel values $\kappa(x, y)$ without explicitly representing the feature vectors. The inner product $\langle x_i, x_j \rangle$ in SVM when replaced by $\kappa(x, y)$ leads to a linear hyperplane in the kernel feature space and a nonlinear one in the original input space. This gives us a tremendous computational advantage for high-dimensional feature spaces.

Let $K(\Re^n)$ denote the class of kernel functions for the mapping $\Re^n \times \Re^n$ to $\Re$. This class is not empty, because $\kappa(x, y) = x^T y$ is a trivial kernel function. The following proposition provides a way for generating additional kernels from an existing kernel.

Proposition 1 $K(\Re^n)$ is closed under addition, multiplication, composition of a continuous function, and addition and multiplication with a positive scalar, i.e. if $\kappa_1, \kappa_2 \in K(\Re^n)$, $\kappa_0 \in K(\Re^m)$ and $\varphi : \Re^n \to \Re^m$ is continuous, then the following five functions again belong to $K(\Re^n)$.

i) $\kappa(x, y) = \kappa_1(x, y) + \kappa_2(x, y)$,

ii) $\kappa(x, y) = \kappa_1(x, y) \cdot \kappa_2(x, y)$,

iii) $\kappa(x, y) = \kappa_1(x, y) + \delta$ for any positive $\delta \in \Re_+$,

iv) $\kappa(x, y) = \kappa_1(x, y) \cdot \delta$ for any positive $\delta \in \Re_+$.

v) $\kappa(x, y) = \kappa_0(\varphi(x), \varphi(y))$.■

For further reading and details of kernel function properties see [3].

Now, we will list the most well-known and useful kernels used in classification tasks in the following table.

http://www.ecology.uni-corvinus.hu ● ISSN 1589 1623

Table 1. *Some well-known kernels*

Gaussian RBF kernel:	$\kappa(x,y)=\exp\left(-\frac{\|x-y\|^2}{\sigma}\right)$, where $\sigma\in\Re_+$
Polynomial kernel:	$\kappa(x,y)=(x^Ty+\sigma)^q$, where $\sigma\in\Re, q\in\mathbf{N}$
Rational quadratic kernel:	$\kappa(x,y)=1-\frac{\|x-y\|^2}{\|x-y\|^2+\sigma}$, where $\sigma\in\Re$
Inverse multi-quadratic kernel:	$\kappa(x,y)=\frac{1}{\sqrt{\|x-y\|^2+\sigma}}$, where $\sigma\in\Re$
Cosine polynomial kernel:	$\kappa(x,y)=\cos\left(\frac{x^Ty}{\|x\|\|y\|}+\sigma\right)^q$, where $\sigma\in\Re, q\in\mathbf{N}$

Weighted kernel functions

When the inner product is computed, one might weight the features such that more important features are given higher weightings than the less important ones. Let $\Re^n$ be an n-dimensional feature space, and $w_1, w_2, \ldots, w_n$ the weights where w_i corresponds to the ith feature. Afterwards the weighted inner product of two vectors $x, y \in \Re^n$ is evaluated in the following way. Let $U = diag(w_1, w_2, \ldots w_n)$ be a diagonal matrix constructed from weights $w_1, w_2, \ldots, w_n$. Then x^TUy is the weighted inner product of vectors x and y.

This idea is also applicable to kernel functions. If κ is a kernel, then $\kappa(Ux,Uy)$ will be a weighted kernel. The following proposition states this in a more general way.

Proposition 2 Let U be a $m \times n$ matrix and $x, y \in \Re^n$ be two vectors. If $\kappa_0 \in K(\Re^n)$ is a kernel, then $k(x,y) = k_0(Ux,Uy)$ is also a kernel function and belongs to $K(\Re^m)$.

Proof. Let the function $\varphi : \Re^n \to \Re^m$ be defined by $\varphi(x) = Ux$. This function is continuous and, because $\kappa_0 \in K(\Re^n)$, $\kappa(x,y) = \kappa_0(\varphi(x),\varphi(y)) = \kappa_0(Ux,Uy)$ is again a valid kernel. This follows from v) of Proposition 1.

Feature ranking methods

Here we describe the feature ranking methods which are used in experiments to rank the features, so that if a feature is more important or less important in a classification then it is ranked accordingly. Most of these methods are traditionally known as feature selection methods because they retain only a small number of features from the top of the ranked list. With this trick, classification will hopefully be speeded up and be more accurate.

http://www.ecology.uni-corvinus.hu ● ISSN 1589 1623

Table 2. *Feature ranking methods used in experiments*

Method name	Description
Std. dev.	The standard deviation method computes the deviation along the direction of the basis vectors. Then it ranks the features according to their values.
Fisher	The Fisher Correlation Coefficient [4] is described in the introduction of this article.
R^2W^2	This method is based on a minimising generalisation bound through gradient descent and is feasible computationally via SVMs. This allows several new possibilities: one can speed up time-critical applications and one can perform feature selection. This method scales the input parameters with a real-valued vector σ, larger values of σ_i indicating more useful features [6]. For further details see [5].
RFE	Recursive Feature Elimination (RFE) is a recently proposed feature selection method described in [7]. The method, given that one wishes to have only $r < n$ input dimensions in the final decision rule, attempts to find the best subset *r*. The method seeks to choose the 'best' *r* features that lead to the largest margin of class separation using an SVM classifier. For each iteration of the training, this combinatorial problem is solved in a greedy fashion by removing those input features that decrease the margin the least until just *r* input features remain. This is known as backward selection [6].
L0	Zero-Norm feature selection can be expressed as the following minimisation problem: $\min_{w \in \Re^n} \quad \|w\|_p$ subject to : $y_i (w \cdot x_i + b) \geq 1$ and $\|w\|_0 \leq r$ where $p = \{1,2\}$ and *r* is the desired number of features. This method can be approximated by minimizing the zero norm using the ℓ_2-AROM or ℓ_1-AROM methods, halting the step-wise minimisation when the constraint $\|w\|_0 \leq r$ is met [6]. One can then re-train a *p*-norm classifier on those features corresponding to the nonzero elements of *w*. In this way one is free to choose the parameter *r* which dictates how many features the classifier will see. This method is based on SVM.
FSV	In the Feature Selection via the concaVe minimisation (FSV) [8] approach, a separating plane is generated by minimising the weighted sum of distances of misclassified points to two parallel planes that bound the set, and which determine the separating plane midway between them. The number of dimensions of the space is used to determine how the plane is minimised. SVM is used in this method.
Entropy	The basic concept of entropy in information theory, first introduced by Shannon, has to do with how much randomness there is in a signal or in a random event. Let $H(f_i) = -p_1 \log_2(p_1) - p_2 \log_2(p_2)$ be the entropy for the *i*th feature, where p_1, p_2 is the rate of the positive and negative examples in the *i*th feature, respectively. The feature ranking is based on these entropy values.

http://www.ecology.uni-corvinus.hu ● ISSN 1589 1623

Experimental Results

We tried out our feature weighting methods on three selected databases, namely two biological datasets of the UCI Machine Learning Repository, the AML/ALL leukaemia dataset at http://lara.enm.bris.ac.uk/colin and the Structural Classification of Proteins (SCOP) database.

We made use of SVM Light as a classification algorithm, and most of the feature ranking methods which are part of SpiderSVM. These algorithms are also available at http://www.kernel-machines.org/software.html and http://www.kyb.tuebingen.mpg.de/bs/people/spider/index.html, respectively. The Standard Deviation and Entropy weighting methods were implemented by us.

To weight the features, the features are first ranked by each ranking method mentioned in the previous section. Afterwards, the ranked features are weighted by the $g(i) = 2^{-i}$ monotonic decrease function, i.e. the *i*th feature of the ranked list is assigned a weight of 2^{-i} except in the Fisher method. With the latter, it has its own feature weighting procedure. The given weights corresponding to features are stored in a diagonal matrix, and it is used for weighting the kernel in the way described above.

In order to measure the performance of these weighted kernel methods we applied an accuracy analysis followed by a receiver operating characteristic (ROC) analysis [11]. The accuracy for a method is simply the fraction of the true predictions of the total number of predictions. The ROC score is the normalized area under a curve that plots sensitivity as a function of specificity for varying classification thresholds. A perfect classifier that puts all the positives at the top of the ranked list will receive an ROC score of 1, while a random classifier will receive an ROC score of 0.5.

Ranking a lot of features naturally requires a lot of time. Hence we apply a dimensionality reduction method called Locally Linear Embedding (LLE) [15]. This is an unsupervised learning method that computes low dimensional, neighbourhood preserving embeddings of high dimensional data. We used this on high dimensional datasets like the leukaemia dataset and SCOP [14]. Details of this method and the Matlab code of the LLE are available at http://www.cs.toronto.edu/~roweis/lle/. Because the LLE preserves the neighbourhoods, the SVM classification does not change significantly because it is based on the inner product of two vectors.

The methods were implemented in Matlab and were run on an IBM PC machine with a 3GHz Intel Pentium IV processor, 4Gbyte RAM and a Windows XP operating system.

Feature weighting and SVM parameters

For the SVM classification we chose the Gaussian RBF kernel function with a σ parameter defined as the median Euclidean distance in the input space from any positive training example to the nearest negative example. The parameter *c* of SVM was set to 1, and a 2-*norm* with value 0.01 was used.

Tests on the AML/ ALL database

The challenge here is to distinguish acute myeloid leukaemia (AML) from acute lymphoblastic leukaemia (ALL). The databases consist of 47 and 25 bone marrow or peripherial blood samples taken from 72 patients of type ALL and AML respectively, with 7129 features per sample, also available at http://lara.enm.bris.ac.uk/colin [9].

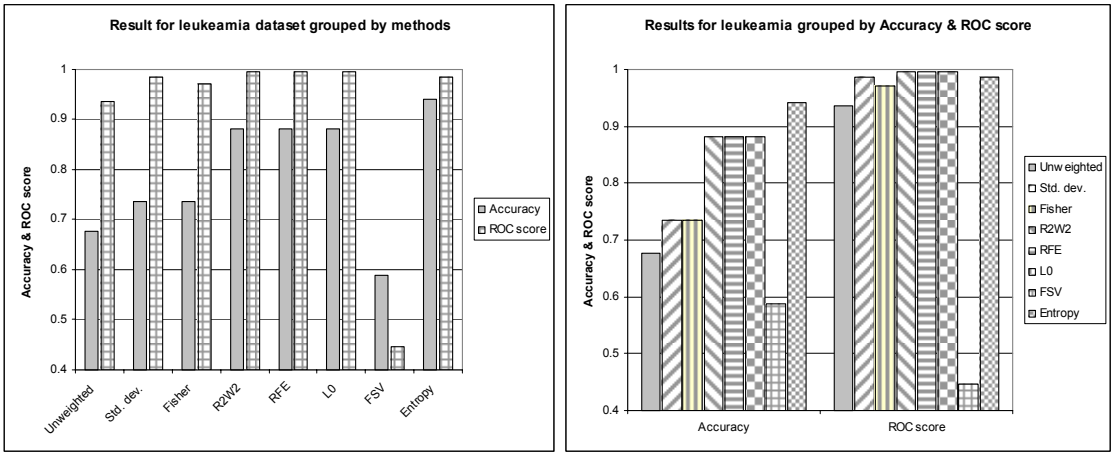
http://www.ecology.uni-corvinus.hu ● ISSN 1589 1623

These samples were produced by AFFymetrix high-density oligonucleotid microarrays [10].

First of all, we reduced the features from 7129 to 10 using LLE. The training set and the test set contained 39 and 34 samples respectively. The results are given in the following table and figures.

Table 3. *Results for the Leukeamia AML-ALL database*

	Unweighted	Std. dev.	Fisher	R^2W^2	RFE	L0	FSV	Entropy
Accuracy	0.67	0.74	0.73	0.88	0.88	0.88	0.59	**0.94**
ROC score	0.93	0.98	0.97	0.99	**1.00**	0.99	0.45	0.99
Time(s)	0.06	0.03	0.04	0.08	0.03	0.08	0.05	0.05



In this database the feature weighting methods perform significantly better than the usual unweighted method. As can be seen, the best accuracy results are given by the feature weighting algorithm based on entropy. We also significantly improved the ROC scores by using a feature weighting method that achieved an ROC score of 1.

Tests on the SCOP databases

The SCOP databases designed by Jakkoola et al. [13] for the remote protein homology are simulated by retaining all members of a target SCOP family from a given superfamily [2]. The sequences were selected using the Astral databases (http://astral.stanford.edu) [14]. Here, positive training examples are chosen from the remaining families in the same superfamily, and negative test and training examples are chosen from disjoint sets of folds outside the target family's fold [12]. Details of the datasets are available at http://www.soe.ucsc.edu/research/compbio/discriminative. The dataset can also be found at http://cs.columbia.edu/compbio/svm-pairvise. In the following *Table 4* we summarise the main details of the dataset used.

Table 4. *Description of the SCOP datasets*

ID	Family name	Positiv train	Negativ train	Positiv test	Negative test	Dimension numbers
SCOP 2.1.1.5	E set domains	94	194	27	39	270
SCOP 2.44.1.2	Eukaryotic proteases	11	14	140	183	25
SCOP 3.32.1.13	Extended AAA-ATPase domain	43	184	8	32	227
SCOP 7.41.5.1	Rubredoxin	10	112	9	98	112

http://www.ecology.uni-corvinus.hu ● ISSN 1589 1623

A numerical database was obtained from this protein sequence database in the following way. A protein sequence A is represented by a vector $F_A = a_1, a_2, \ldots, a_n$, where n is the number of training proteins, and a_i is the similarity score of A and A_i proteins by Smith- Watermann algorithm, as implemented on the BioXLP hardware accelerator (www.cgen.com).

Before we carried out the classification procedure, every high-dimensional dataset was reduced to an 8-dimensional input space, except for the SCOP 2.44.1.2 dataset because its dimensionality was low.

Table 5. *Results for the SCOP datasets*

	Unweighted	Std. dev.	Fisher	R^2W^2	RFE	L0	FSV	Entropy
	SCOP 2.1.1.5							
Accuracy	0.66	0.68	**0.75**	0.67	0.67	0.67	0.68	0.67
ROC score	0.79	**0.89**	0.88	0.85	0.84	0.84	0.86	0.85
Time(s)	7.2	7.9	7.9	8.0	8.0	7.8	6.3	8.4
	SCOP 2.44.1.2							
Accuracy	**0.59**	0.48	0.52	0.54	0.54	0.54	0.51	0.49
ROC score	0.35	0.14	**0.72**	0.47	0.47	0.47	0.70	0.59
Time(s)	0.13	0.13	0.13	0.14	0.13	0.14	014	0.13
	SCOP 3.32.1.13							
Accuracy	0.86	0.86	0.81	**0.89**	**0.89**	**0.89**	0.86	0.86
ROC score	0.85	0.92	0.85	**0.93**	**0.93**	**0.93**	0.87	0.81
Time(s)	3.0	2.8	2.0	2.5	2.6	2.5	2.4	1.9
	SCOP 7.41.5.1							
Accuracy	0.92	0.92	0.92	0.92	0.92	0.92	0.92	0.92
ROC score	0.61	0.80	**0.84**	0.75	0.75	0.75	0.74	0.73
Time(s)	0.53	0.64	0.64	0.66	0.66	0.64	0.58	0.43

With these datasets better results were indeed obtained by applying feature weighting methods. The ROC scores achieved by using weighted feature methods were significantly better than those for the usual unweighted case. The accuracy scores using the weighted methods were also better than the baseline but no significant improvement was obtained. This may be because there were too many dimensions and too few training and test examples for a learning method to work effectively.

Tests on the UCI Machine Learning Repository

The UCI Machine Learning Repository is a database that contains millions of records and thousands of field types widely used in business, medicine, engineering, and the sciences. This datasets is available at http://www.ics.uci.edu/~mlearn. We chose the Heart and Hepatitis biological and medical databases, which are listed in *Table 6*. These databases were not divided into train and test sets originally, so we used 10-fold cross validation for testing.

Table 6. *Description of the heart and Hepatitis UCI datasets*

Dataset name	# Features	#Instances	Class #1 name	Class #2 name
Heart	13	270	Absence of heart disease	Presence of heart disease
Hepatitis	19	155	Die	Live

http://www.ecology.uni-corvinus.hu ● ISSN 1589 1623

The performance of the feature weighting methods on each of these datasets is presented in the table below. In these experiments the baseline unweighted methods provided the best results. A reason for this might be because all of these features were equally important or that another weighting function should be applied here.

Table 7. *Results for the UCI datasets*

	Unweighted	**Std. dev.**	**Fisher**	**R^2W^2**	**RFE**	**L0**	**FSV**	**Entropy**
				HEART				
Accuracy	**0.84**	0.79	0.82	0.76	0.76	0.76	0.81	0.77
ROC score	**0.91**	0.85	0.88	0.85	0.85	0.85	0.89	0.85
Time(s)	23.0	6.7	7.5	8.5	8.0	8.0	8.9	6.7
				HEPATITIS				
Accuracy	**0.83**	0.79	0.81	0.76	0.76	0.76	0.80	0.80
ROC score	**0.92**	0.85	0.89	0.82	0.82	0.82	0.83	0.86
Time(s)	4.9	1.6	1.2	3.3	3.2	3.3	2.0	3.1

Conclusions and further work

Here we introduced a feature weighting method for SVMs that is based on a feature ranking methodology. To enable us to do this, several feature ranking procedures were applied, and then the weights were assigned to these ranked features. In some real-world biological classification experiments we showed that we could obtain more accurate predictions using this method, and in a very short time.

In this paper we focused on feature rankings using the weighting function $g(i) = 2^{-i}$. This, of course, is unsuitable in high dimensional features spaces which may be of order ten/hundred/thousand, because most of the features are assigned almost zero weights. On the other hand, the importance of certain features could not be represented exactly by this weighting function as the importance of ranked features did not, for instance, decrease in quite the same way as the weighting function. Further study is needed to learn the effect of our approach on special databases and to find out whether other good methods exist that allow us to determine the weights for each ranked feature.

REFERENCES

[1] Bosser, B.E., Guyon, M., Vapnik, V.N. (1992): A Training Algorithm for Optimal Margin Classifiers. - Proc. Of the Fifth Annual ACM Conference on Computational Learning Theory

[2] Leslie, C., Eskin, E., Cohen, A., Weston, J., Stafford Noble, W. (2004): Mismatch string kernels for discriminative protein classification. - Bioinformatics

[3] Berg, C., Christensen, J., Ressel, P. (1984): Harmonic Analysis on Semigroups: Theory of Positive Definite and Related Functions. - Springer

[4] Bishop, C. (1995): Neural Networks for Pattern Recognition. - Oxford UP, Oxford, UK

[5] Weston, J., Mukherjee, S., Chapelle, O., Pontil, M., Poggio, T., Vapnik, V. (2001): Feature Selection for SVMs, Advances in Neural Information Processing Systems 13. - MIT Press, Cambridge, MA, USA,

[6] Weston, J., Elisseff, A., Schölkopf, B., Tipping, M. (2003): Use of the zero-norm with linear models and kernel methods. - Journal of Machine Learning Research

http://www.ecology.uni-corvinus.hu ● ISSN 1589 1623

[7] Guyon, I., Weston, J., Branhill, S., Vapnik, V. (2002): Gene Selection for Cancer Classification using Support Vector Machines. - Machine Learning 46.
[8] Bradley, P.S., Mangasarian, O.L. (1998): Feature Selection via Concave Minimization and Support Vector Machines. - INFORMS Journal on Computing 10.
[9] Shevade, S.K., Keerthi, S.S., (2002): A Simple and Efficient Algorithm for Gene Selection using Sparse Logistic Regression. - Bioinformatics
[10] Furey, T.S., Cristianini, N., Duffy, N., Bednarski, D.W., Schummer, M., Haussler, D. (2000): Support Vector Machine Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data. - Bioinformatics 16.
[11] Gribskov, M., Robinson, N. (1996): The Use of Receiver Operating Characteristic (ROC) Analyisis to Evaluate Sequence Matching. - Computers and Chemistry 20
[12] Lialo, L., Noble, W.S. (2003): Combining Pairwise Sequence Similarity and Support Vector Machines for Detecting Remote Protein Ecolutionary and Structural Relationships. - Journal of Computational Biology
[13] Jaakkola, T., Diekhans, M., Haussler, D. (2000): A discriminative framework for detecting remote protein homologies. - Journal of Computational Biology
[14] Brenner, S.S., Koehl, P., Levitt, M. (2000): The ASTRAL compendium for sequence and structure analysis. - Nucl. Acad. Sci. 28 USA.
[15] Saul, L., Roweis, S.T. (2000): Nonlinear dimensionality reduction by locally linear embedding. - Science 290

http://www.ecology.uni-corvinus.hu ● ISSN 1589 1623