ESANN 2013 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 24-26 April 2013, i6doc.com publ., ISBN 978-2-87419-081-0. Available from http://www.i6doc.com/en/livre/?GCOI=28001100131010.

A new metric for dissimilarity data classification based on Support Vector Machines optimization

Agata Manolova[1] and Anne Guerin-Dugue[2]

1- Technical University Sofia - Faculty of Telecommunications
8 ave Kliment Ohridski, Sofia 1000 - Bulgaria
2- GIPSA-Lab - Department of Images and Signal
Grenoble INP-CNRS - France

Abstract. Dissimilarities are extremely useful in many real-world pattern classification problems, where the data resides in a complicated, complex space, and it can be very difficult, if not impossible, to find useful feature vector representations. In these cases a dissimilarity representation may be easier to come by. The goal of this work is to provide a new technique based on Support Vector Machines (SVM) optimization that can be a good alternative in terms of accuracy compared to known methods using dissimilarities such as *k* nearest neighbor classifier (*k*NN), prototype-based dissimilarity classifiers and distance kernel based SVM classifiers.

1 Introduction

Pattern recognition is traditionally based on a representation with features. Features should preferably be defined on the basis of expert knowledge of the application domain. So each object is represented by a vector in a multidimensional feature space.

In recent years, Pekalska, Duin and others have proposed alternative representations of the observations by using dissimilarities [1]. According to them, if we assume that the objects called "similar" can be grouped to form a class, a "class" is nothing more than a collection of these similar objects. Based on this idea, Duin and colleagues argue that the notion of proximity (similarity or dissimilarity) is actually more fundamental to define a class than features [1]. The advantage of dissimilarity-based classifiers is that since they do not operate on the class-conditional distributions, their accuracy can exceed theoretically the Bayes' error bound. Also they do not have to confront the problems associated with feature spaces such as the "curse of dimensionality", and the issue of estimating a large number of parameters. The use of dissimilarities to represent objects opens new possibilities in statistical learning, for the dissimilarities can capture both the statistical and structural information about objects.

The most popular dissimilarity-based classifier is *k* nearest neighbor (*k*NN). It requires no prior knowledge about the distribution of the data. It is fast and simple. But it is not well suited to measures that do not respect the triangular inequality.

The support vector machines (SVMs) are widely used in statistical learning for classification and regression. SVM are kernel-based methods. The notion of similarity (and thus dissimilarity) is closely related to the use of kernels. Classical kernels are defined using scalar products (similarity) or Euclidean distance (dissimilarity) between feature vectors in the initial feature space. They are defined in vector spaces such as to meet the Mercer's conditions [3]. Some problem specific non-metric (e.g. violate the triangle inequality) distance measures often lead to kernels which are not positive definite. Such kernels are based on tangent-distance, dynamic-time-warping distance or Kullback-Leibler divergence [1, 2, 4]. This type of SVM with distance kernels can also be interpreted as optimal hyper plane classifier [2]. To our knowledge, there are three approaches, proposed for the use of dissimilarity data in SVMs. The first is Pekalska's prototype selection method presented in [1]. The second method involves multidimensional scaling of the dissimilarity matrix and classification of the data in this space with linear SVMs [1]. The third approach is inspired by the use of distance kernels for specific classification problems [4].

This paper focuses on the incorporation of SVM in to the dissimilarity-based metric "Shape Coefficient" (*Cs*), described in details in [5]. The *Cs* is defined from simple statistics (mean and variance) on the dissimilarity data. The proposed decision rule is based on the *Cs* description and on the optimal separating hyper plane with Support Vector Machines (SVM). This provides a decision rule with a limited number of parameters per class.

The article is organized as follows: in Section 2 we describe the theoretical basis of this approach; in Section 3 we provide experimental results on real-life data sets. Finally, Section 4 concludes the article.

2 Description of the "Shape Coefficient" metric and the decision rule based on SVM optimization procedure

2.1 The "Shape coefficient" metric

In this work, we used the dissimilarity representation space inspired by Pekalska's work and we define after recoding the dissimilarity data, a model which was parameterized using a low-dimensional parameter space.

Let us consider a two-class classification problem where ω_1 is the first class and ω_2 the second class. Let *N* be the number of objects o_i in a set to be classified, *D* is the dissimilarity (*N*×*N*) matrix between each object such as: $D=[d(o_{i,}o_j):1\le i,j\le N]$. The logic of the construction of the *Cs* follows the reasoning of the discriminant analysis -maximizing the inter-class inertia and minimizing the intra-class inertia to best separate the class. Following [5], the metric *Cs* describes the proximity of an object to a given class (for example for ω_1):

$$Cs(o_i,\omega_1)=\gamma_1\frac{\left(\overline{d^2(o_i,\omega_1)}-I(\omega_1)\right)^2}{\left(\mathrm{var}\left\{d^2(o_i,\omega_1)\right\}\right)^{\delta_1}}, \qquad (1)$$

where $\overline{d^2(o_i,\omega_1)}$ is the empirical average of the dissimilarity between object o_i and all the observations in class ω_1, $\mathrm{var}\{d^2(o_i,\omega_1)\}$ is the empirical variance, and $I(\omega_1)$ is the class inertia computed as the empirical mean of all the squared dissimilarities between objects in class ω_1. The numerator deals with the "position" of the observation o_i relatively the class center. The denominator interpretation is more complex, taking into account the "structure" (orientation, shape, intrinsic dimension…) of the observations distribution in the class. In order to adapt the *Cs* to different types of class distributions, the learning parameters δ_1 and γ_1 are added to best fit this data structure. The equation for $Cs(o_i ,\omega_2)$ with the class ω_2 is equivalent to (1) and has two fitting parameters γ_2 and δ_2. The decision rule for a two-class classification problem for an object o_i is given then by the following relation:

$$\text{class}(o_i) = \begin{cases} \omega_1, \text{if } Cs(o_i,\omega_1) < Cs(o_i,\omega_2) \\ \omega_2, \text{if } Cs(o_i,\omega_1) > Cs(o_i,\omega_2) \end{cases} \quad (2)$$

This relation is defined by 4 learning parameters for a two-class problem. From (2), we have defined in [5] learning procedures which overcame or have had similar performance compared to the *k*NN rule. In the next subsection we present a more straightforward learning procedure based on SVM optimization with only three independent parameters: δ_1, δ_2, γ_1/γ_2.

2.2 Decision rules using SVM procedure

The idea is to propose a new representation of the observations which must be compatible with a linear decision rule in this new features space. The quantities $Cs(o_i ,\omega_1)$ and $Cs(o_i ,\omega_2)$ being positive, we can transform (2) using the logarithmic function as follows:

$$\begin{aligned} &\log(\gamma_1/\gamma_2) + 2\log(\overline{d^2(o_i,\omega_1)} - I(\omega_1)) - 2\log(\overline{d^2(o_i,\omega_2)} - I(\omega_2)) \\ &-\delta_1 \log(\text{var}(d^2(o_i,\omega_1))) + \delta_2 \log(\text{var}(d^2(o_i,\omega_2))) \begin{cases} < 0 \text{ if } o_i \in \omega_1 \\ > 0 \text{ if } o_i \in \omega_2 \end{cases} \end{aligned} \quad (3)$$

In fact this (3) can be interpreted as such a rule in a four dimensional vector space. It represents a separating hyper plane separating the two classes when we replace the inequality with equality. Following (3) we can represent each object o_i using a four dimensional feature vector $\mathbf{x}_i$ ($\mathbf{x}_i = [x_{i1}\, x_{i2}\, x_{i3}\, x_{i4}]^T$) by adopting the recoding of the variables in (3):

$$\begin{aligned} x_{i1} &= 2 \times \log(\overline{d^2(o_i,\omega_1)} - I(\omega_1)) & x_{i3} &= -\log(\text{var}(d^2(o_i,\omega_1))) \\ x_{i2} &= -2 \times \log(\overline{d^2(o_i,\omega_2)} - I(\omega_2)) & x_{i4} &= \log(\text{var}(d^2(o_i,\omega_2))) \end{aligned} \quad (4)$$

By adopting the usual vector notation, equation (3) becomes: $\boldsymbol{\beta}^T \mathbf{x}_i + \beta_0 \underset{\omega_2}{\overset{\omega_1}{\lessgtr}} 0$,

where $\boldsymbol{\beta}$ = $[1\ 1\ \delta_1\ \delta_2]^T$, is the normal to the optimal separating hyper plane and $\beta_0 = \log(\gamma_1/\gamma_2)$ is the bias from the hyper plane to the origin. Labeling the objects with the auxiliary variables per class, such as y_i = -1 for $o_i \in \omega_1$ and y_i = 1 for $o_i \in \omega_2$, we have the classical linear decision rule: $\hat{y}_i = sign(\boldsymbol{\beta}^T \mathbf{x}_i + \beta_0)$. This is the standard decision rule for classical SVM. Here, the difference with the classical solution concerns the vector $\boldsymbol{\beta}$ normal to the optimal hyper plane: in this case it is constraint to have the same two first components: $\beta_1 = \beta_2$.

We chose to use the SVM optimization problem, since the theory of the proposed metric does not assume special properties of the class distributions. We have to modify it for learning in the context of this partial knowledge of the normal to the separating hyper plane. To do this, we group the feature vectors $\mathbf{x}_i$ into two orthogonal subspaces:

$$\mathbf{x}_i = \begin{bmatrix} \mathbf{x}'_i \\ \mathbf{x}''_i \end{bmatrix} \text{with } \mathbf{x}'_i = \begin{bmatrix} x_{i1} \\ x_{i2} \end{bmatrix} \text{and } \mathbf{x}''_i = \begin{bmatrix} x_{i3} \\ x_{i4} \end{bmatrix} \quad \boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}' \\ \boldsymbol{\beta}'' \end{bmatrix} \text{with } \boldsymbol{\beta}' = \|\boldsymbol{\beta}'\| \mathbf{1}_N \text{ and } \boldsymbol{\beta}'' = \begin{bmatrix} \beta_3 \\ \beta_4 \end{bmatrix},$$

where the vector's $\mathbf{1}_N$ norm is 1 and : $\mathbf{1}_N = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \Big/ \sqrt{2}$.

And by using the dual optimization problem, (3) can be interpreted as a function to minimize with 4 unknown parameters: $\|\boldsymbol{\beta}'\|$, β_3, β_4 and β_0. This problem consists in finding the optimal hyper plane when the 2 classes are non separable and it is solved by using the Lagrange multipliers [3]. In our case, and using the reasoning presented in the above paragraph the optimization problem can be rewritten as follows:

$$\begin{aligned} &\text{Minimize}_{\|\boldsymbol{\beta}'\|, \boldsymbol{\beta}'', \beta_0, \boldsymbol{\xi}} \quad \frac{1}{2} \|\boldsymbol{\beta}'\|^2 + \frac{1}{2} \|\boldsymbol{\beta}''\|^2 + C \textstyle\sum_{i=1}^{N} \xi_i \\ &\text{Subject to}: y_i(\|\boldsymbol{\beta}'\| u_i + \boldsymbol{\beta}''^T \mathbf{x}''_i + \beta_0) \geq 1 - \xi_i, \text{ for } i = 1, \ldots, N \text{ and } \xi_i \geq 0, \\ &\text{where } u_i = \langle \mathbf{1}_N ; \mathbf{x}'_i \rangle \end{aligned} \tag{5}$$

where ξ_i are the slack variables, associated with all the objects, for misclassified object ξ_i >1. The parameter C corresponds to the penalty for errors and it is chosen by the user. Using the solution with Lagrange multipliers and the Karush-Kuhn-Tucker conditions for the primal problem this gives the dual Lagrangian to minimize, where α_i are the Lagrange multipliers:

$$\begin{aligned} &\text{Minimize } L_D = -\sum_i \alpha_i + \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \left(u_i u_j + \langle \mathbf{x}''_i ; \mathbf{x}''_j \rangle \right) \\ &\text{Subject to}: 0 \leq \alpha_i \leq C \text{ and } \sum_i \alpha_i y_i = 0 \end{aligned} , \tag{6}$$

Therefore, we end up with the same dual Lagrangian (6) as the SVM with the difference of the term $u_i \cdot u_j$ added to the scalar product $\langle \mathbf{x}_i ; \mathbf{x}_j \rangle$ for each pair of observations.

The classifier presented above aims to find the optimal separating hyper plane knowing that the $\boldsymbol{\beta}$ vector normal to the optimal hyper plane is forced to have two identical components: $\beta_1 = \beta_2$ and the feature space of this hyper plane is limited to four dimensions regardless of the underlying intrinsic dimensions of the observations, and the number of observations.

3 Experimental results on real-life dissimilarity datasets

The dissimilarity databases used in this research are public and available in the Internet. They are described in Table 1 and analyzed following the same methodology as in [2, 6]. To characterize the databases, we calculate a negative eigenratio (NER) [6]. It is the ratio of the largest negative to the largest positive eigenvalue of the dissimilarity matrix. NER is a measure of non-Euclidean behavior of the dissimilarity. If the distance measure is almost Euclidean the NER is small (<0.1). These different datasets represent a wide spectrum from easily to difficultly separable data. None of the dissimilarity measures are isometric to an L^2-norm. The estimate of the recognition rate for these databases is achieved using "Leave One Out" (LOO), due to the rather small size of the databases and for the multi-class SVM we use the “one vs. all” procedure. The method *Cs*-SVM, was developed in C, by introducing some modifications to the open source software SVM^{Light} 6.02, in order to implement our dual Lagrange formulation incorporating a priori knowledge on the $\boldsymbol{\beta}$ vector.

Name	Number of classes/[Number of objects]	Dissimilarity measure	NER
Kimia 1/Kimia2	6/[12 per class]	Hausdoff distance	0.05/0.1
Music-EMD1/2	4/ [22 28 27 30]	Earth Mover's Distance	0.41/0.48
Music-PTD1/2	4 /[22 28 27 30]	Proportional Transportation Distance	0.31/0.28
UNIPEN-DWT	5/[50 per class]	Dynamic Time Warping	0.2
USPS-TD1,2,3,4	2/ [146 104], [133 117], [169 81], [160 90]	Tangent distance	0.07

Table 1: Collection of dissimilarity databases used in this study

In Table 2 are presented the best results in LOO classification error for all the datasets, the values of the penalty error *C* and the parameter for the Gaussian radial basis kernel k^{rbf} are logarithmically varying along a suitable grid according to [2]. The recognition error rates for the pure distance substitution linear kernel (k^{lin}) and k^{rbf} kernel are taken directly from [2], thus these classifiers have not been re-implemented and we used their best estimates. We have chosen to compare *Cs*-SVM with the k^{rbf} kernel because it exhibits the best recognition error behavior according to [2].

Name	*k*NN	k^{lin} SVM	k^{rbf} SVM	*Cs*-SVM
Kimia 1/Kimia2	6.94/16.67	15.28/12.50	4.17/9.72	6.49/16.67
Music-EMD1/2	26/29.82	40.00/42.11	20.00/10.53	2.00/8.77
Music-PTD1/2	36/38.60	34.00/31.58	32.00/28.07	4.00/10.53
UNIPEN-DWT	6/7.60	14.40/10.80	5.20/6.00	4.40/5.60
USPS-TD1, 2	4.40/5.20	10.40/14.40	3.20/2.40	3.20/2.80
USPS-TD3, 4	4.40/2.80	12.80/10.80	4.00/3.20	1.20/1.20

Table 2: LOO recognition error [%] of the classification experiments

These experiments demonstrate the effectiveness of the proposed classifier compared to some of the existing dissimilarity-based classifiers. *Cs*-SVM is successful in most cases even when NER is high. Example of the comparison with SVM k^{rbf} is shown on fig. 1 where the axes represent the classification errors for the k^{rbf} classifier and respectively *Cs*-SVM for each database. For every point below the median, *Cs*-SVM performs better than the k^{rbf}.

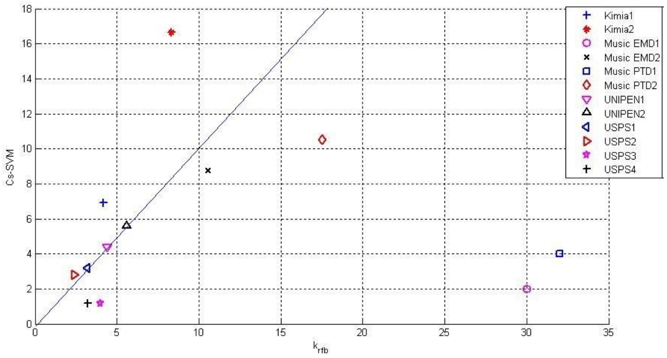

Fig. 1: LOO error rates from Table 2 for different datasets for k^{rbf} and *Cs*-SVM.

4 Conclusions and further research

The classifier proposed in this work improves the error recognition results for most datasets compared to the pure distance substitution k^{lin} and k^{rbf} kernel SVM. *Cs*-SVM is rather simple and requires little setup because its performance does not depend on the choice of an appropriate distance kernel. It is very robust for different types of dissimilarities. Results on real datasets show that *Cs*-SVM is a good alternative to other dissimilarity-based classifiers because of its: (a) parsimony (only based on first and second order statistics on dissimilarity values), (b) datadriven flexibility (two fitting parameters to learn the “shape” and the “intrinsic dimension” of each class) and (c) stable behavior when facing incomplete dissimilarity data.

For our future research we would like to explore the possibility to use the primal optimization problem that can also be solved efficiently for non-linear SVM but compared to the dual problem the first can be rewritten as an unconstrained problem.

References

[1] E. Pękalska, Robert P.W. Duin. *The dissimilarity representation for pattern recognition. Foundations and Applications*, World Scientific, Singapore, December 2005.

[2] B. Haasdonk, C. Bahlmann. Learning with Distance Substitution Kernels. Pattern Recognition - *Proc. of the 26th DAGM Symposium*, Tübingen, Germany, pp. 220-227. Springer Berlin, 2004.

[3] N. Cristianini, J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*, Cambridge University Press, 2000.

[4] B. Haasdonk, E.Pękalska. Classification with Kernel Mahalanobis Distance Classifiers, *German Classification Society Annual Conference*, 2008.

[5] A. Manolova, G. Celeux, A.Guerin-Dugue. Classification of dissimilarity data with a new flexible Mahalanobis like metric, *PAA Special Issue on Non-parametric Distance-based Classification Techniques and their Applications*, Springer-link , 11(3-4): 337-351, 2008.

[6] R.P.W. Duin and E. Pękalska, Datasets and tools for dissimilarity analysis in pattern recognition, Technical Report 2009, SIMBAD (EU,FP7,FET), 2009, 1-174.