ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 355-360

1-v-1 Tri-Class SV Machine

Cecilio Angulo[a] and Luis González[b]

[a]Technical University of Catalonia, GREC Research Group,
08800, Vilanova i la Geltrú, Spain, cangulo@esaii.upc.es
[b]University of Seville, Applied Economics I Dept.,
41018, Sevilla, Spain, luisgon@us.es

Abstract. The 1-v-1 tri-class SV machine is specially addressed to avoid the loss of information that occurs in the usual 1-v-1 training procedure, meanwhile a similar two-phases (decomposition, reconstruction) scheme is used. The new machine presents all the advantages of 1-v-1 training and it allows to incorporate by means of a tri-class scheme all the information contained into the training patterns when a multi-class problem is considered.

1 Introduction

Support Vector Machines are learning machines implementing the structural risk minimization inductive principle to obtain good generalization on a limited number of learning patterns. This theory was originally developed by Vapnik on the basis of a separable binary classification problem with signed outputs ± 1 [9].

SVM presents good theoretical properties and behavior in problems of binary classification [3]. There exist many works generalizing the original bi-class approach to multi-classification problems ([7], [8]) through different algorithms, like 1-v-r SVM or 1-v-1 SVM. This paper improves the original idea developed in [2] and [1].

The paper is organized as follows, in section 2, the standard SVM classification learning paradigm is briefly presented in order to introduce some notations. Section 3 is devoted to a short introduction to SVMs for multi-classification. In Section 4, the 1-v-1 Tri-class SV Machine is presented and some examples illustrate its behavior. Finally, some concluding remarks are displayed.

2 Bi-Class SV Machine Learning

The SV Machine is an implementation of a more general regularisation principle known as the large margin principle. Let $\mathcal{Z} = (x, y) = ((x_1, y_1), \ldots, (x_n, y_n)) = (z_1, \ldots, z_n) \in (\mathcal{X} \times \mathcal{Y})^n$ be a training set, with $\mathcal{X}$ being the input space and $\mathcal{Y} = \{\theta_1, \theta_2\} = \{-1, +1\}$ the output space. Let $\phi : \mathcal{X} \to \mathcal{F} \subseteq \mathbb{R}^d$, with $\phi = (\phi_1, \ldots, \phi_d)$, be a feature mapping for the usual 'kernel trick'. $\mathcal{F}$ is named *feature space*. Let $\mathbf{x} \stackrel{def}{=} \phi(x) \in \mathcal{F}$ be the *representation* of $x \in \mathcal{X}$. A (binary) linear classifier, $f_{\mathbf{w}}(x) = \langle \phi(x), \mathbf{w} \rangle = \langle \mathbf{x}, \mathbf{w} \rangle$, is searched for in the space $\mathcal{F}$, with $f_{\mathbf{w}} : \mathcal{X} \to \mathcal{F} \to \mathbb{R}$, and outputs

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 355-360

will be obtained by thresholding it, $h_{\mathbf{w}}(x) = \text{sign}(f_{\mathbf{w}}(x))$. The classifier $\mathbf{w}$ with the largest geometrical margin on a given training sample $\mathcal{Z}$ can be written as, [6]

$$\mathbf{w}_{SVM} \stackrel{def}{=} \arg\max_{\mathbf{w}\in\mathcal{F}} \frac{1}{\|\mathbf{w}\|} \cdot \min_{z_i\in\mathcal{Z}} y_i\langle \mathbf{x_i}, \mathbf{w}\rangle \tag{1}$$

A method computationally amenable of casting the problem is to minimize the norm $\|\mathbf{w}\|$ in (1) with the geometrical margin fixed to unity

$$\begin{aligned} \min_{\mathbf{w}\in\mathcal{F}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{subject to} \quad & y_i\langle \mathbf{x_i}, \mathbf{w}\rangle \geq 1 \quad z_i \in \mathcal{Z} \end{aligned} \tag{2}$$

Solution is

$$\mathbf{w}_{SVM} = \sum_i \alpha_i\, y_i\, \mathbf{x_i} \quad ; \quad f_{\mathbf{w}_{SVM}}(x) = \sum_i \alpha_i\, y_i\, k(x_i, x) \tag{3}$$

where $k(x, x') = \langle \phi(x), \phi(x')\rangle = \langle \mathbf{x}, \mathbf{x}'\rangle$ is the kernel function, and only a few α_i are not zero, those associated to the so-called *support vectors*.

3 SV Machine for Multi-Classification

Let Z be a training set. Now, a set of possible labels $\{\theta_1, \cdots, \theta_\ell\}$, with $\ell > 2$ will be considered. Subsets $Z_k \in Z$, defined as $Z_k = \{z_i = (x_i, y_i) : y_i = \theta_k\}$ generate a partition in Z. It will be denoted $n_k = \#Z_k$, so $n = n_1 + n_2 + \cdots + n_\ell$. If I_k is the number of index i being $z_i \in Z_k$, it follows $\bigcup_{i\in I_k} \{(\mathbf{x}_i, y_i)\} = Z_k$.

A very usual multi-classification SVM approach is 1-v-1 SVM: a first decomposition phase generates several learning machines in parallel, each machine having in consideration only two classes, and a reconstruction scheme allows to obtain the overall output by merging outputs from the decomposition phase. In this approach, $\frac{\ell\cdot(\ell-1)}{2}$ binary classifiers are trained to generate hyperplanes f_{kh}, $1 \leq k < h \leq \ell$, separating training vectors Z_k with label θ_k from training vectors in class θ_h, Z_h. If f_{kh} discriminates without error then $sign(f_{kh}(\mathbf{x}_i)) = 1$, for $z_i \in Z_k$ and $sign(f_{kh}(\mathbf{x}_i)) = -1$, for $z_i \in Z_h$. Remaining training vectors $Z \setminus \{Z_k \bigcup Z_h\}$ are not considered in the optimisation problem. Hence, for a new entry x, numeric output from each machine $f_{kh}(x)$ is interpreted as,

$$\Theta(f_{kh}(x)) = \begin{cases} \theta_k & \text{if } sign(f_{kh}(x)) = 1 \\ \theta_h & \text{if } sign(f_{kh}(x)) = -1 \end{cases} \tag{4}$$

In the reconstruction phase, labels distribution generated by the trained machines in the parallel decomposition is considered through a merging scheme.

The 1-v-1 multi-classification approach is usually preferred to the 1-v-r scheme [7]. Main drawback for this approach is that only data from two classes is considered to train each machine, so output variance is high and any information from the rest of classes is ignored.

If a hyperplane f_{kh} must classify an input x_i with $i \notin I_k \bigcup I_h$, only output $f_{kh}(x_i) = 0$ will not be translated into an incorrect interpretation. The natural improvement to be analysed is to force every training input in different classes to θ_k and θ_h to be contained into the separating hyperplane $f_{kh}(\mathbf{x}) = 0$.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 355-360

In [1] a novel procedure is presented where remaining training vectors are forced to be encapsulated into a δ-tube, $0 \leq \delta < 1$, along the separation hyperplane. Parameter δ allows to create a slack zone (a 'tube') around the hyperplane where remaining training vectors are covered. The separating hyperplane must solve the optimisation problem,

$$\begin{aligned} \min_{\mathbf{w}\in\mathcal{F}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C_1 \cdot \sum_i \xi_i + C_2 \cdot \sum_j (\varphi_j + \varphi_j^*) \\ \text{subject to} \quad & \begin{cases} y_i \langle \mathbf{w}, \mathbf{x_i} \rangle \geq 1 - \xi_i & z_i \in \mathcal{Z}_{1,2} \\ -\delta - \varphi_j^* \leq \langle \mathbf{w}, \mathbf{x_j} \rangle \leq \delta + \varphi_j & z_j \in \mathcal{Z}_3 \\ \xi_i \geq 0 & z_i \in \mathcal{Z}_{1,2} \\ \varphi_j^*, \varphi_j \geq 0 & z_j \in \mathcal{Z}_3 \end{cases} \end{aligned} \tag{5}$$

where $\mathcal{Z}_{1,2}$ are the patterns belonging to the classes labelled as $\{-1, +1\}$ and $\mathcal{Z}_3$ are those labelled with 0. The solution has a similar form to (3), being α_i the multipliers associated to the problem, accomplishing $\sum_i \alpha_i = 0$. For a new entry x, the numeric output from the machine $f_{\mathbf{w}}(x)$ is interpreted as

$$\Theta(f_{\mathbf{w}}(x)) = \begin{cases} 1 & \text{if } f_{\mathbf{w}}(x) > \delta \\ -1 & \text{if } f_{\mathbf{w}}(x) < -\delta \\ 0 & \text{if } |f_{\mathbf{w}}(x)| \leq \delta \end{cases} \tag{6}$$

This approach has demonstrated good results on standard 'benchmarks' [1], but in the general case, it is necessary to select many parameters[1]: (i) k, kernel function; (ii) C_1, associated weight for the sum of errors into the two discriminated classes; (iii) C_2, associated weight for the sum of errors into the remaining classes; (iv) δ, insensitivity parameter.

4 1-v-1 Tri-class SVM

The number of tuning parameters can be reduced if the margin to be maximized in (2) is that defined between the patterns assigned with output $\{-1, +1\}$ and the entries labelled with 0, the remaining patterns. In this case, the width of the 'decision tube' along the decision hyperplane where 0-labeled patterns are allocated is no considered 'a priori' and the δ parameter is eliminated. A classifier with this characteristic must to accomplish

$$\mathbf{w}_{SV3} \stackrel{def}{=} \arg\max_{\mathbf{w}\in\mathcal{F}} \frac{1}{\|\mathbf{w}\|} \cdot \left\{ \min_{z_i \in \mathcal{Z}_{1,2}} y_i \langle \mathbf{x_i}, \mathbf{w} \rangle - \max_{z_i \in \mathcal{Z}_3} |\langle \mathbf{x_i}, \mathbf{w} \rangle| \right\} \tag{7}$$

When $\|\mathbf{w}\|$ is minimized while the rest of the product is fixed to unitary distance, (7) can be translated into

$$\begin{aligned} \min_{\mathbf{w}\in\mathcal{F}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 \\ \text{subject to} \quad & y_i \langle \mathbf{x_i}, \mathbf{w} \rangle \geq 1 + |\langle \mathbf{x_j}, \mathbf{w} \rangle| \qquad z_i \in \mathcal{Z}_{1,2}; \quad z_j \in \mathcal{Z}_3 \end{aligned} \tag{8}$$

in a more amenable manner[2].

[1] An extended study can be found in [4].
[2] Constraints are a bit stronger than (7).

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 355-360

This optimisation problem is consistent with the standard formulation because if all the 0-labeled training patterns were exactly on the decision hyperplane (i.e. no incorrect interpretation is possible) or these patterns were not concerning into the problem, then the novel machine is similar to the 1-v-1 SVM machine.
Restrictions can be relaxed to allow little errors with the ± 1-labeled training patterns by using 'slack' variables

$$\xi_i = 1 + \max_{z_j \in \mathcal{Z}_3} |\langle \mathbf{x_j}, \mathbf{w} \rangle| - y_i \langle \mathbf{x_i}, \mathbf{w} \rangle \geq 0 \qquad z_i \in \mathcal{Z}_{1,2} \tag{9}$$

and restrictions in (8) can be manipulated to obtain the optimisation problem

$$\begin{array}{ll} \min\limits_{\mathbf{w} \in \mathcal{F}} & \frac{1}{2}\|\mathbf{w}\|^2 + C \sum\limits_i \xi_i \\ \text{s.t.} & \left\{ \begin{array}{ll} y_i \langle \mathbf{x_i} - \mathbf{x_j}, \mathbf{w} \rangle - 1 + \xi_i \geq 0 & z_i \in \mathcal{Z}_{1,2};\ z_j \in \mathcal{Z}_3 \\ \xi_i \geq 0 & z_i \in \mathcal{Z}_{1,2} \end{array} \right. \end{array} \tag{10}$$

When Lagrange multipliers are applied to the original optimization problem, it is obtained

$$\mathcal{L} = \frac{1}{2}\|\mathbf{w}\|^2 + C \sum_i \xi_i + \sum_{ij} \alpha_{ij} \left(1 - \xi_i - y_i \langle \mathbf{x_i} - \mathbf{x_j}, \mathbf{w} \rangle\right) - \sum_i \mu_i \xi_i \tag{11}$$

with

$$0 \leq \sum_j \alpha_{ij} \leq C, \quad z_i \in \mathcal{Z}_{1,2}; \qquad \mathbf{w} = \sum_{ij} y_i\, \alpha_{ij}\, (\mathbf{x_i} - \mathbf{x_j}) \tag{12}$$

The dual problem is,

$$\begin{array}{ll} \max\limits_{\alpha} & \sum\limits_{i,j} \alpha_{ij} - \sum\limits_{ijkl} y_i y_k \alpha_{ij} \alpha_{kl} \langle \mathbf{x_i} - \mathbf{x_j}, \mathbf{x_k} - \mathbf{x_l} \rangle \\ \text{subject to} & \left\{ \begin{array}{ll} 0 \leq \sum\limits_j \alpha_{ij} \leq C & \\ \alpha_{ij}, \alpha_{kl} \geq 0, & z_i, z_k \in \mathcal{Z}_{1,2}; \quad z_j, z_l \in \mathcal{Z}_3 \end{array} \right. \end{array} \tag{13}$$

and the solution function can be written,

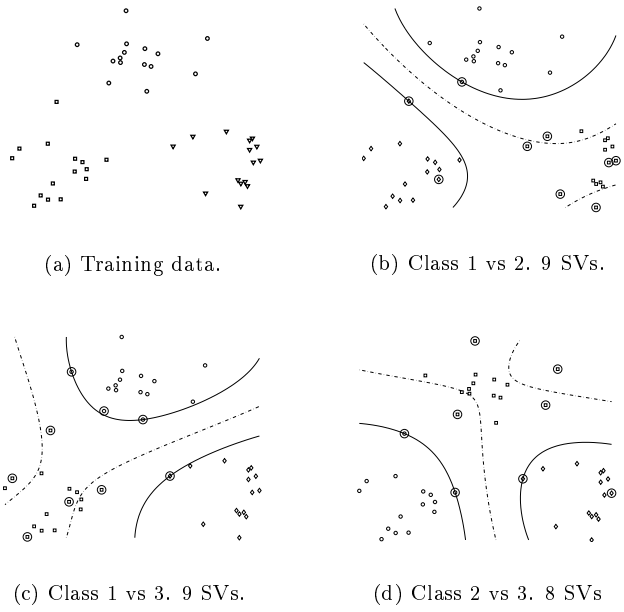
$$f_{\mathbf{w}}(x) = \sum_{ij} \alpha_{ij} y_i \left(k(x_i, x) - k(x_j, x)\right) \tag{14}$$

For a new entry x, output is interpreted according to (6), where

$$\delta = \max_{z_j \in \mathcal{Z}_3} |f_{\mathbf{w}}(x_j)| = \max_{z_j \in \mathcal{Z}_3} |\langle \mathbf{w}, x_j \rangle| \tag{15}$$

In Figure 1 the behavior of the novel machine is illustrated on a simple separable problem with gaussian kernel. Support vectors (SVs) are those patterns with null associated parameters, i.e. a null row or column in the parameter matrix. As it was expected, the number of support vector is limited and they lie in the margin. Solid lines are indicating the δ-tube for the 'remaining vectors' and dot line is the separating hyperplane. It must be noted that values for δ are very low, in this example 0.1126, 0.1750 and 0.2159.
In Figure 2 the performance of the novel machine is showed when it is applied with gaussian kernel on a multi-class no separable problem. It can be observed that a little band between classes remains unclassified because the outputs from the parallel decomposition phase assign this zone to different classes.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 355-360



(a) Training data. (b) Class 1 vs 2. 9 SVs.

(c) Class 1 vs 3. 9 SVs. (d) Class 2 vs 3. 8 SVs

Figure 1: Results on a simple separable problem with 45 patterns.

5 Conclusions and Future Works

In this paper, a new kernel machine has been designed to solve multi-classification problems. The machine allows to incorporate by means of a tri-class scheme all the information contained into the training patterns when a multi-class problem is considered. Information from 'remaining patterns' is captured into a δ-tube, where δ is an optimal parameter automatically obtained by maximizing the margin between classes.

Example on the artificial data set show the good performance of this novel machine, and it must be evaluated on standard 'benchmarks'.

New research lines can be started about theoretical generalization bounds of this machine. By observing the constraints in the optimisation problem, a direct extension to ordinal regression problems is being investigated. ECOC methodology can be applied when the machine be evaluated on 'benchmarks'. Moreover, an initiated line is the probabilistic interpretation of the outputs according to their value [5].

6 Acknowledgements

This study was partially supported by Junta de Andalucía grant ACC-265-TIC-2001, and Spanish MCyT grant TIC2002-04371-C02-01.

ESANN'2003 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 23-25 April 2003, d-side publi., ISBN 2-930307-03-X, pp. 355-360

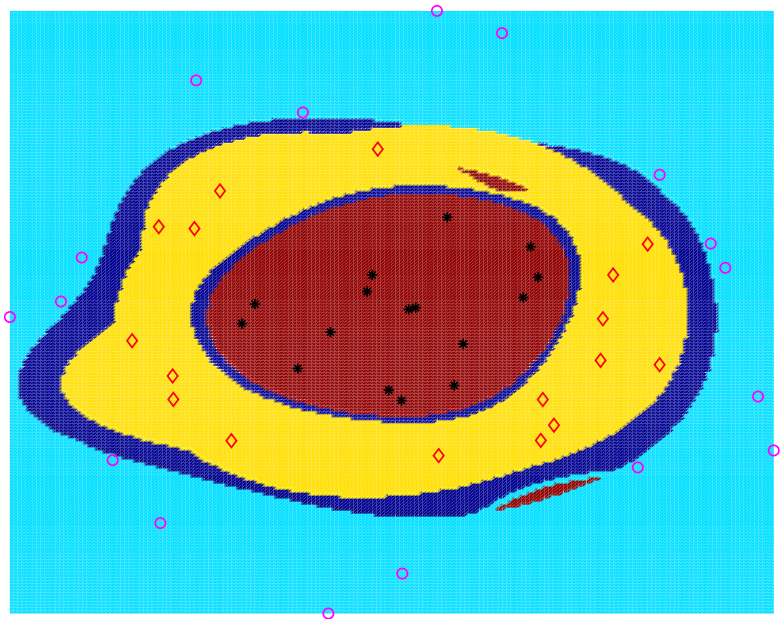


Figure 2: Complete classification on a no separable problem with 50 patterns.

References

[1] C. Angulo. *Learning with Kernel Machines into a Multi-Class Environment.* Doctoral thesis, Technical University of Catalonis, April 2001. In Spanish.

[2] C. Angulo and A. Català. A multi-class support vector machine. *Lecture Notes in Computer Science*, 1810:55–64, 2000.

[3] N. Cristianini and J. Shawe-Taylor. *An introduction to Support Vector Machines and other kernel-based learning methods.* Cambridge University press 2000, 2000.

[4] L. González. *Análisis discriminante utilizando máquinas núcleos de vectores soporte. Función núcleo similitud.* Tesis doctoral, Universidad de Sevilla, Marzo 2002.

[5] L. González, C. Angulo, F. Velasco, and M. Vilchez. Máquina ℓ-SVCR con salidas probabilísticas. *Inteligencia Artificial*, (17):72–82, september 2002.

[6] R. Hebrich. *Learning Kernel Classifiers. Theory and Algorithms.* The MIT Press, 2002.

[7] U. Kressel. Pairwise classification and support vector machine. *In B. Schölkopf, C. Burgues and A. Smola, editors*, Advances in Kernel Methods: support Vector Learning. MIT Press. Cambridge, MA, 1999.

[8] E. Mayoraz and E. Alpaydin. Support vector machines for multi-class clasification. *Proc. of the IWANN*, 1999.

[9] V.N. Vapnik. *Statistical Learning Theory.* John Wiley & Sons, Inc, 1998.