INCORPORATING PRIOR INFORMATION INTO SUPPORT VECTOR MACHINES IN THE FORM OF ELLIPSOIDAL KNOWLEDGE SETS

Jean-Baptiste Pothin, Cédric Richard

Institut des Sciences et Technologies de l'Information de Troyes (ISTIT-M2S, FRE CNRS 2732)
Université de Technologie de Troyes, 12 rue Marie Curie, BP 2060, 10010 Troyes cedex - France
jean_baptiste.pothin@utt.fr cedric.richard@utt.fr

ABSTRACT

This paper investigates a learning model in which the training set contains prior information in the form of ellipsoidal knowledge sets. We handle this problem in a minimax setting, which consists of maximizing the worst-case – minimum – margin between the knowledge sets from the two classes and the decision surface. The problem is solved using an alternating optimization scheme and an active learning strategy, i.e., the training set is created progressively according to the prior information. Our approach is evaluated on toy examples and on a usual benchmark database. It is successfully compared to state-of-the-art techniques.

1. INTRODUCTION

Support Vector Machines (SVMs) are widely considered to be among the best performing algorithms for supervised classification. Their success is due to the following ideas: Firstly, data are mapped into a high dimensional space where the classes of data are more readily separable. Secondly, margin – or distance – between the separating hyperplane and the closest points of each class is maximized.

While it has been well established in the field of machine learning that incorporating prior knowledge helps a classifier achieving more accurate generalization, little work has been done to incorporate prior into SVMs, see e.g. [1]. In this paper, we are interested in prior information in the form of ellipsoidal knowledge sets. They consist of labelled regions in the input space, and can thus be interpreted as a generalization of the usual notion of training example. In [2], the authors investigate a learning model in which the observed input $\boldsymbol{x}$ is corrupted by additive uniformly distributed noise. They incorporate this uncertainty in the form of spheroidal knowledge sets, and develop a kernel-based algorithm to solve it. This approach is called TSVC (Total Support Vector Classification), by reference to the total least-squares method to which it is related. However, in TSVC, only the furthest points from the separating hyperplane are considered to determine the latter, leading to a solution in the best-case framework. Spheroidal knowledge sets are also considered in [3]. They result from a data reduction process, called Set Covering SVM (SC-SVM), used to enhance the training process of SVMs when dealing with large data sets. However, the authors reduce the problem to that of the classification of the clusters centers with a usual SVM. Finally, prior knowledge in the form of ellipsoidal constraints is incorporated into a semidefinite linear program in [4]. However, no kernel-based extension of this work is proposed.

In this paper, we overcome all these limits by reformulating the problem in a minimax setting. It consists of maximizing the worst-case – minimum – margin between the knowledge sets from the two classes and the decision surface. We also investigate a kernel-based extension of this approach. The remainder of this paper is organized as follows. In Section 2, the minimax knowledge-based approach for optimizing two-class SVMs is presented. An effective means of resolving this problem is described in Section 3, and tested experimentally in Section 4. Finally, concluding remarks and some suggestions for further studies follow.

2. TRAINING SVM WITH ELLIPSOIDAL KNOWLEDGE SETS

Before presenting the contributions of this paper, we give a brief overview of SVM learning for classification. Let $\mathcal{A}_n = \{(\boldsymbol{x}_i, y_i) \in (\mathcal{X} \times \mathcal{Y})\}_{i=1}^n$ be a n-sample training set, with $\mathcal{X}$ the input space and $\mathcal{Y} = \{\pm 1\}$ the label set. Training a L$_1$-SVM is finding the hyperplane $\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = 0$ that satisfies:[1]

$$\begin{array}{ll} \min_{\boldsymbol{w},\boldsymbol{\xi},b} & \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_i \xi_i \\ \text{subject to} & y_i(\langle \boldsymbol{w}, \boldsymbol{x}_i \rangle + b) \geq 1 - \xi_i, \xi_i \geq 0 \end{array} \qquad (1)$$

for $i = 1, \ldots, n$. In the above equations, ξ_i is the i-th slack variable among n, and C a positive parameter controlling the trade-off between margin maximization and error min-

[1] All sums run from 1 to n, unless otherwise noted.

imization. To solve this optimization problem, one introduces its dual form:

$$\begin{array}{cll} \min_{\boldsymbol{\alpha}} & \frac{1}{2}\sum_i\sum_j \alpha_i\alpha_j y_i y_j \langle \boldsymbol{x}_i, \boldsymbol{x}_j\rangle - \sum_i \alpha_i & \\ \text{subject to} & \sum_i \alpha_i y_i = 0 & (2) \\ {\scriptstyle 1\le i\le n} & 0 \le \alpha_i \le C. & \end{array}$$

The solution $\boldsymbol{w}$ can be expressed as $\boldsymbol{w} = \sum_i \alpha_i y_i \boldsymbol{x}_i$, and the decision function is $d(\boldsymbol{x}) = \text{sgn}(\sum_i \alpha_i y_i \langle \boldsymbol{x}_i, \boldsymbol{x}\rangle + b)$.

In this paper, we consider the problem of incorporating prior knowledge into SVM training in the form of ellipsoidal knowledge sets. That is, we assume that the training set consists of n labelled ellipsoids $(\mathcal{E}_i, y_i)$ defined as[2]

$$\mathcal{E}_i = \{\boldsymbol{x} \in \mathcal{X} \mid \langle \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x} - \overline{\boldsymbol{x}}_i), (\boldsymbol{x} - \overline{\boldsymbol{x}}_i)\rangle \le 1\}, \qquad (3)$$

where $\overline{\boldsymbol{x}}_i$ is the center of the i-th ellipsoid and $\boldsymbol{\Sigma}_i$ is a symmetric positive (semi)-definite matrix. Henceforth, we shall denote by $\mathcal{A}_n$ the training set $\{(\boldsymbol{x}, y_i) : \boldsymbol{x} \in \mathcal{E}_i\}_{i=1}^n$.

2.1. Hard-Margin Ellipsoid Machine (HMEM)

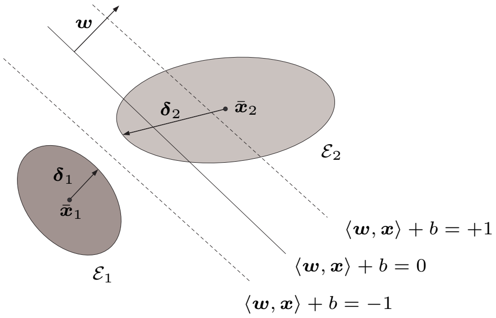
Let us consider the case where the training set is linearly separable. Then there exist functions $f(\boldsymbol{x}) = \langle \boldsymbol{w}, \boldsymbol{x}\rangle + b$ such that $y\, f(\boldsymbol{x}) > 0$ for all $(\boldsymbol{x}, y)$ in $\mathcal{A}_n$. A special case is obtained by scaling $\boldsymbol{w}$ and b such that training data closest to the decision surface satisfy $|\langle \boldsymbol{w}, \boldsymbol{x}\rangle + b| = 1$. This canonical form satisfies $y\, f(\boldsymbol{x}) \ge 1$ for all $(\boldsymbol{x}, y)$ in $\mathcal{A}_n$. Combining the latter with (3) yields the conditions

$$y_i\left(\langle \boldsymbol{w}, \overline{\boldsymbol{x}}_i + \boldsymbol{\delta}_i\rangle + b\right) \ge 1, \text{ with } \langle \boldsymbol{\Sigma}_i^{-1}\boldsymbol{\delta}_i, \boldsymbol{\delta}_i\rangle \le 1, \qquad (4)$$

for $i = 1, \ldots, n$. Among the hyperplanes $\langle \boldsymbol{w}, \boldsymbol{x}\rangle + b = 0$ satisfying (4), we look for the one with the maximum distance from the decision surface to the closest points from the two classes. This distance, called the margin, is known to be equal to $1/\|\boldsymbol{w}\|$. The optimal hyperplane then maximizes, over $\boldsymbol{w}$ and b, the minimum margin with respect to $\boldsymbol{\delta}_1, \ldots, \boldsymbol{\delta}_n$ and subject to (4). Note that maximizing (resp., minimizing) $1/\|\boldsymbol{w}\|$ is equivalent to minimizing (resp., maximizing) $\|\boldsymbol{w}\|^2$. Thus, we can solve the following problem to determine the optimal hyperplane

$$\begin{array}{cll} \min_{\boldsymbol{w},b}\max_{\boldsymbol{\delta}_i\text{'s}} & \frac{1}{2}\|\boldsymbol{w}\|^2 & \\ \text{subject to} & y_i\left(\langle \boldsymbol{w}, \overline{\boldsymbol{x}}_i + \boldsymbol{\delta}_i\rangle + b\right) \ge 1 & (5) \\ {\scriptstyle 1\le i\le n} & \langle \boldsymbol{\Sigma}_i^{-1}\boldsymbol{\delta}_i, \boldsymbol{\delta}_i\rangle \le 1. & \end{array}$$

This minimax strategy consists of maximizing the worst-case – minimum – margin between the knowledge sets from the two classes. It contrasts with the TSVC method [2], which reduces to solving the standard SVM problem with the furthest points in each spheroidal knowledge set from the separating hyperplane. It also differs from the SC-SVM method [3], which incorporates knowledge sets into the problem formulation via their centers. The performances of these approaches are experimentally compared in Section 4.



$\boldsymbol{w}$
$\boldsymbol{\delta}_2$
$\bar{\boldsymbol{x}}_2$
$\mathcal{E}_2$
$\boldsymbol{\delta}_1$
$\bar{\boldsymbol{x}}_1$
$\mathcal{E}_1$
$\langle \boldsymbol{w}, \boldsymbol{x}\rangle + b = +1$
$\langle \boldsymbol{w}, \boldsymbol{x}\rangle + b = 0$
$\langle \boldsymbol{w}, \boldsymbol{x}\rangle + b = -1$

Fig. 1. Most critical points and slack variables. The ellipsoidal knowledge set $\mathcal{E}_1$ is correctly classified. The slack variables of the $\boldsymbol{x}_i$'s in $\mathcal{E}_1$ are zero. The most critical point $\bar{\boldsymbol{x}}_2 + \boldsymbol{\delta}_2$ of $\mathcal{E}_2$ is in the margin. Its slack variable is the largest one, for all $\boldsymbol{x}_i \in \mathcal{E}_2$.

2.2. Soft-Margin Ellipsoid Machine (SMEM)

Consider now the case in which training data cannot be separated without error. Following the standard practice (1) for SVM, we relax the hard-margin constraints (4) with slack variables, and we modify the objective function to penalize the violation of these constraints. This leads to the soft-margin problem

$$\begin{array}{cll} \min_{\boldsymbol{w},b}\max_{\boldsymbol{\delta}_i\text{'s}} & \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_i \xi_i & \\ \text{subject to} & y_i\left(\langle \boldsymbol{w}, \bar{\boldsymbol{x}}_i + \boldsymbol{\delta}_i\rangle + b\right) \ge 1 - \xi_i & \\ {\scriptstyle 1\le i\le n} & \langle \boldsymbol{\Sigma}_i^{-1}\boldsymbol{\delta}_i, \boldsymbol{\delta}_i\rangle \le 1, & (6) \\ & \xi_i \ge 0. & \end{array}$$

Resolution of this problem is discussed in the next section. Of course, other penalization terms on the empirical error, such as $\sum_i \xi_i^2$ for instance [5], should also be used.

3. ALTERNATING OPTIMIZATION FOR SMEM

Suppose that the points associated with the largest slack variables $\xi_i > 0$ are known, see Fig. 1. These are denoted by $\boldsymbol{x}_i^c$ to make reference to the fact that they are the most critical ones. In that case, problem (6) can be solved by replacing $\boldsymbol{x}_i$ with $\boldsymbol{x}_i^c$ in the formulation (2). The idea is then to train a SMEM as follows. First, we fix the parameters $\boldsymbol{w}$ and b of the separating hyperplane in order to determine the most critical samples $\boldsymbol{x}_i^c$. Note that they can be computed analytically, as shown below. Next, we find the separating hyperplane by solving the standard SVM problem (2) with the critical points as a training set. This two-step process is repeated until convergence, see Table 1.

To avoid situations where the location of the separating hyperplane changes in a cyclic fashion, we suggest to use both present and past critical samples $\boldsymbol{x}_i^c$ for training. Since this strategy suffers from the disadvantage that the size of the training set continuously increases, we suggest to use a sparsification process to discard redundant data: any critical point $\boldsymbol{x}_i^c$ is included in the training set if, and only if, it

[2] A natural extension of our work is to consider hybrid training sets of ellipsoids and data samples. This is an easy exercice left to the reader.

```
Initializations: $\mathcal{A}_n^{(0)} = \{\bar{\boldsymbol{x}}_i, y_i\}_{i=1}^n$, $k = 0$
Repeat
    Use a SVM algorithm with $\mathcal{A}_n^{(k)}$ to get $(\boldsymbol{w}, b)$
    Determine $\boldsymbol{x}_i^c$ with (11), for every ellipsoid $\mathcal{E}_{i=1,\dots,n}$
    Construct $\mathcal{A}_n^{(k+1)}$ by inserting into $\mathcal{A}_n^{(k)}$, the $\boldsymbol{x}_i^c$'s that satisfy
        the novelty condition: $\|\boldsymbol{x}_i^c - \boldsymbol{x}_j\| > \epsilon, \forall \boldsymbol{x}_j \in \mathcal{A}_n^{(k)}$
Until $\mathcal{A}_n^{(k+1)} \equiv \mathcal{A}_n^{(k)}$
Return $(\boldsymbol{w}, b)$ and the support vectors
```

Table 1. SMEM algorithm in the linear kernel case

satisfies $\|\boldsymbol{x}_i^c - \boldsymbol{x}_j\| > \epsilon$ for all $\boldsymbol{x}_j$ in the current training set. We established that this condition guarantees the finiteness of the training set. Due to lack of space, this result will be presented in a companion paper.

3.1. Most critical points in the linear case

According to the constraints in (6), for fixed separating hyperplane $\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = 0$, the point $\boldsymbol{x}$ of $\mathcal{E}_i$ which minimizes $y_i(\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b)$ is associated with the largest slack variable. Thus, to find the most critical point of $\mathcal{E}_i$, we have to solve[3]

$$\boldsymbol{\delta}_i = \arg\min_{\boldsymbol{\delta}} \; y_i(\langle \boldsymbol{w}, (\bar{\boldsymbol{x}}_i + \boldsymbol{\delta}) \rangle + b) \qquad (7)$$

subject to $\langle \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta}, \boldsymbol{\delta} \rangle \leq 1$. By omitting constant terms, the functional to be minimized becomes $y_i \langle \boldsymbol{w}, \boldsymbol{\delta} \rangle$. Since the inequality constraint is active at the optimum, the Lagrangian of (7) can be expressed as:

$$L(\boldsymbol{\delta}, \lambda) = y_i \langle \boldsymbol{w}, \boldsymbol{\delta} \rangle + \lambda(\langle \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta}, \boldsymbol{\delta} \rangle - 1). \qquad (8)$$

The optimality conditions of (8) being

$$\begin{array}{lll} \frac{\partial L(\boldsymbol{\delta},\lambda)}{\partial \boldsymbol{\delta}} = 0 & \Rightarrow & y_i \boldsymbol{w} + 2\lambda \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta} = 0 \\ \frac{\partial L(\boldsymbol{\delta},\lambda)}{\partial \lambda} = 0 & \Rightarrow & \langle \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta}, \boldsymbol{\delta} \rangle = 1, \end{array} \qquad (9)$$

by combining (9.a) and (9.b) we get:

$$\boldsymbol{\delta}_i = -y_i \frac{\boldsymbol{\Sigma}_i \boldsymbol{w}}{\sqrt{\langle \boldsymbol{\Sigma}_i \boldsymbol{w}, \boldsymbol{w} \rangle}}. \qquad (10)$$

The most critical point is thus:

$$\boldsymbol{x}_i^c = \bar{\boldsymbol{x}}_i - y_i \frac{\boldsymbol{\Sigma}_i \boldsymbol{w}}{\sqrt{\langle \boldsymbol{\Sigma}_i \boldsymbol{w}, \boldsymbol{w} \rangle}}. \qquad (11)$$

If our task was to find the least critical point, we would have to maximise $y_i(\langle \boldsymbol{w}, (\bar{\boldsymbol{x}}_i + \boldsymbol{\delta}) \rangle + b)$. In that case, the solution is of the same form as (11) with the sign of $\boldsymbol{\delta}_i$ reversed. Geometrically, the most and the least critical points define a straight line with $\bar{\boldsymbol{x}}_i$ in the middle, $\boldsymbol{x}_i^c - \bar{\boldsymbol{x}}_i$ being related to the so-called discriminative direction [6]. Note that the authors of the TSVC method formulate the problem of classifying hyperspheres as a standard SVM problem [2]. This leads them to treat each hypersphere as its least critical point since the margin reaches its maximum value in this case. A consequence of this is that the TSVC approach does not necessarily result in a zero-error separating hyperplane when the data is linearly separable, whereas our algorithm perfectly separates both classes. Due to lack of space, this result cannot be proved here.

3.2. Most critical points in the nonlinear case

By using a kernel function $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle$, the data is implicitly mapped to a reproducing kernel Hilbert space $\mathcal{F}$, called the feature space. This operation, which transforms the regular manifold $\mathcal{E}_i$ of $\mathcal{X}$ into an unknown one $\phi(\mathcal{E}_i)$ of $\mathcal{F}$, makes our algorithm inapplicable. To overcome this difficulty, note that every critical point should minimize the following function over $\boldsymbol{\delta}$, see (7):

$$y_i \langle \boldsymbol{w}, \phi(\bar{\boldsymbol{x}}_i + \boldsymbol{\delta}) \rangle, \qquad (12)$$

subject to $\langle \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta}, \boldsymbol{\delta} \rangle \leq 1$. A curve line analysis in the input space locates the most critical points on the surface of the ellipsoids [7], meaning that $\langle \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta}, \boldsymbol{\delta} \rangle = 1$ is likely to be true. Therefore, by using the dual expansion of $\boldsymbol{w}$, the problem can be written as:

$$\boldsymbol{\delta}_i = \arg\min_{\boldsymbol{\delta}} \; y_i \sum_j \alpha_j y_j \kappa(\bar{\boldsymbol{x}}_i + \boldsymbol{\delta}, \boldsymbol{x}_j) \qquad (13)$$

subject to $\langle \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta}, \boldsymbol{\delta} \rangle = 1$. If we now assume the differentiability of κ, the first order Taylor expansion

$$\kappa(\bar{\boldsymbol{x}}_i + \boldsymbol{\delta}, \cdot) \simeq \kappa(\bar{\boldsymbol{x}}_i, \cdot) + \langle \nabla_{\boldsymbol{x}} \kappa(\boldsymbol{x}, \cdot)|_{\boldsymbol{x}=\bar{\boldsymbol{x}}_i}, \boldsymbol{\delta} \rangle \qquad (14)$$

leads to the optimization problem

$$\boldsymbol{\delta}_i = \arg\min_{\boldsymbol{\delta}} \; y_i \sum_j \alpha_j y_j \langle \nabla_{\boldsymbol{x}} \kappa(\boldsymbol{x}, \boldsymbol{x}_j)|_{\boldsymbol{x}=\bar{\boldsymbol{x}}_i}, \boldsymbol{\delta} \rangle \qquad (15)$$

subject to $\langle \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta}, \boldsymbol{\delta} \rangle = 1$. The Lagrangian $L(\boldsymbol{\delta}, \lambda)$ is

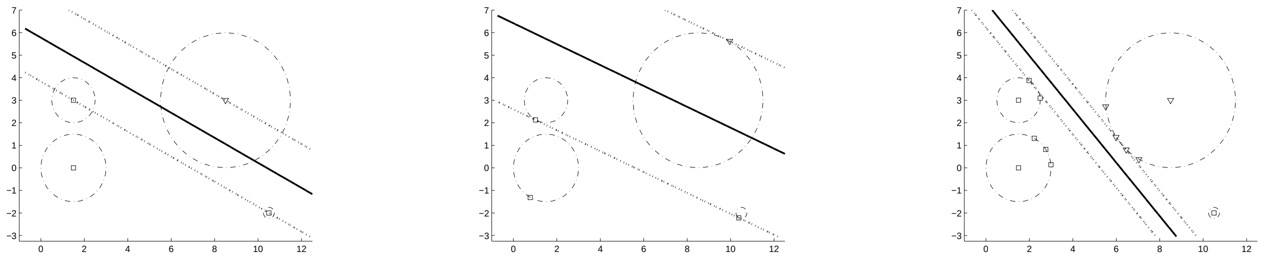
$$\begin{aligned} L(\boldsymbol{\delta}, \lambda) = y_i \sum_j \alpha_j y_j \langle \nabla_{\boldsymbol{x}} \kappa(\boldsymbol{x}, \boldsymbol{x}_j)|_{\boldsymbol{x}=\bar{\boldsymbol{x}}_i}, \boldsymbol{\delta} \rangle \\ + \lambda(\langle \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\delta}, \boldsymbol{\delta} \rangle - 1). \end{aligned} \qquad (16)$$

By combining the optimality conditions for $L(\boldsymbol{\delta}, \lambda)$, we get

$$\boldsymbol{x}_i^c = \bar{\boldsymbol{x}}_i - y_i \frac{\boldsymbol{\Sigma}_i \boldsymbol{v}_i}{\sqrt{\langle \boldsymbol{\Sigma}_i \boldsymbol{v}_i, \boldsymbol{v}_i \rangle}}, \qquad (17)$$

with $\boldsymbol{v}_i = \sum_j \alpha_j y_j \nabla_{\boldsymbol{x}} \kappa(\boldsymbol{x}, \boldsymbol{x}_j)|_{\boldsymbol{x}=\bar{\boldsymbol{x}}_i}$. Finally, our kernel-based SMEM algorithm is obtained by replacing (11) with equation (17) in Table 1, and by using the kernelized novelty condition: $\sqrt{\kappa(\boldsymbol{x}_i^c, \boldsymbol{x}_i^c) + \kappa(\boldsymbol{x}_j, \boldsymbol{x}_j) - 2\kappa(\boldsymbol{x}_i^c, \boldsymbol{x}_j)} > \epsilon$.

[3] Even when the ellipsoid $\mathcal{E}_i$ is correctly classified, i.e., $\xi_i = 0$, minimizing $y_i(\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b)$ leads to the nearest point from the separating hyperplane. This agrees with the notion of criticality.



SC-SVM TSVC SMEM

Fig. 2. Toy example solved with the linear kernel: the biggest circular knowledge set vs. the three others. The samples produced by the algorithms to compute the separating hyperplanes are represented by squares and triangles.

4. EXPERIMENTS

We shall now illustrate our approach using two toy examples. Figure 2 compares SC-SVM and TSVC methods with our SMEM algorithm on a non-linearly separable problem: the biggest circular knowledge set vs. the three others. The linear kernel $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \langle \boldsymbol{x}, \boldsymbol{x}' \rangle$ was selected for this experiment. We clearly observe that the SMEM-based classifier outperforms the others. This result is confirmed by the error rates, estimated to be 11.66%, 11.68% and 0.5% for SC-SVM, TSVC and SMEM, respectively. Figure 3 illustrates the ability of our approach to separate two classes of ellipsoidal knowledge sets with the second-degree polynomial kernel $\kappa(\boldsymbol{x}, \boldsymbol{x}') = (1 + \langle \boldsymbol{x}, \boldsymbol{x}' \rangle)^2$.

Let us turn now to a usual benchmark, the breast cancer Wisconsin database. It consists of 699 patterns, with 9 numerical attributes per pattern. These data were centered, normalized, and randomly divided into a training set of 466 instances and a test set of 233 instances. Spheroidal knowledge sets were created from the training set with the set covering algorithm [3]. They were used to train a SMEM and a TSVC classifier with $\epsilon = 1$ and $C = 1$. The gaussian kernel, $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \exp(\|\boldsymbol{x} - \boldsymbol{x}'\|^2/2\sigma_0^2)$, was selected. Its bandwidth σ_0 was set to 5. The generalization performance of each classifier was estimated using the test set, and averaged over 50 runs. Estimated error rates of $3.56\% \pm 0.01\%$ and $5.68\% \pm 0.07\%$ were obtained for the SMEM and the TSVC classifiers, respectively. This illustrates the superiority of our approach.

5. CONCLUSION

In this paper, we investigated a learning model in which the training set contains prior information in the form of ellipsoidal knowledge sets. We addressed this problem by maximizing the minimum margin between the knowledge sets from the competing classes. Our experiments using this minimax strategy showed substantial performance improvements over the SC-SVM and the TSVC methods. A direct extension of this work is given by the one-class SVM used for novelty detection. Further work includes the potential application of our approach to deal with uncertainty in the observations of a classification problem, that is, to deal with situations where instead of labelled samples we may only have distributions over them.

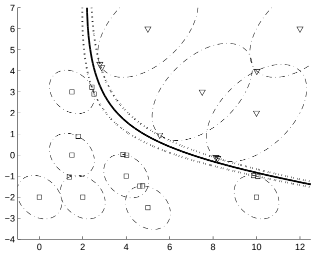


Fig. 3. Toy example solved with our algorithm and the second-degree polynomial kernel.

6. REFERENCES

[1] D. Decoste and B. Schölkopf, “Training invariant support vector machines,” *Machine Learning*, vol. 46, pp. 161–190, 2002.

[2] J. Bi and T. Zhang, “Support vector classification with input data uncertainty,” in *Proc. Advances in Neural Information Processing Systems*, L. K. Saul, Y. Weiss, and L. Bottou, Eds. Cambridge, MA: The MIT Press, 2004.

[3] J. Wang and C. Zhang, “Support vector machines based on set covering,” in *Proc. of the 2nd International Conference on Information Technology for Application*, 2004, pp. 181–184.

[4] V. Jeyakumar, J. Ormerod, and R. S. Womersley, “Knowledge-based semidefinite linear programming classifiers,” *Optimization Methods and Software (to appear)*.

[5] C. Cortes and V. Vapnik, “Support vector networks,” *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[6] P. Golland, “Discriminative direction for kernel classifiers,” *Neural Information Processing Systems*, vol. 13, pp. 745–752, 2001.

[7] S. Akaho, “Svm that maximizes the margin in the input space,” *Systems and Computers in Japan*, vol. 35, no. 14, pp. 78–86, 2004.