First International Conference
Modelling and Development of Intelligent Systems
Sibiu - Romania, 22-25 October, 2009

A comparative analysis on the potential of SVM and k-means in solving classification tasks

Luminiţa State, Iuliana Paraschiv-Munteanu

Abstract

The method based on support vectors aims to increase the efficiency in approximating multidimensional functions. The basic idea in a SVM approach is twofold. On one hand it aims to determine a classifier that minimizes the empirical risk, that is to encode the learning sequence as good as possible with respect to a certain architecture, and the other hand to improve the generalization capacity by minimizing the generalization error. In case of non-linear separable data the SVM is combined with kernel based technique which transforms the data in a linear separable data by mapping the initial data on to higher dimensional space of features. This mapping is performed in terms of special tailored kernels that allow to keep the computations at a reasonable complexity level.

The aim of the research reported in the paper is to obtain an alternative approach in using SVM in case of non-linearly separable data based on using the k-means algorithm instead of the standard kernel based approach. The potential of the proposed approach is pointed out on experimental basis in the final section of the paper. The tests were performed on data generated from multi-dimensional normal repartitions yielding to linearly separable and non-linearly separable samples respectively. The results encourage the research toward integrating the k-means technique in a SVM-based learning scheme.

1 Introduction

The Support Vector Machine (SVM) is a relatively new concept in machine learning and it was introduced by Vapnik ([10], [11]). In designing a classifier, two main problems have to be solved, on one hand the option concerning a suitable structure and on the other hand the selection of an algorithm for parameter estimation.

The algorithm for parameter estimation performs the optimization of a convenable selected cost function with respect to the empirical risk which is directly related to the representativeness of the available learning sequence. The choice of the structure is made such that to maximize the generalization capacity, that is to assure good performance in classifying new data coming from the same classes. In solving these problems one has to establish a balance between the accuracy in encoding the learning sequence and the generalization capacities because usually the over-fitting prevents the minimization of the empirical risk.

2 Supervised learning using SVM

Let us assume that the data is represented by examples coming from two categories or classes such that the true provenance class for each example is known. We refer such a collection of individuals as being a

supervised learning sequence, and it is represented as

$$\mathcal{S}=\left\{ (x_i,y_i) \mid x_i=(x_{i1},\ldots,x_{id})^T \in \mathbf{R}^d,\ y_i \in \{-1,1\},\ i=\overline{1,N} \right\}. \tag{1}$$

The values 1, -1 are taken as labels corresponding to the classes. We say the data is linearly separable if there exists a linear discriminant function $g:\mathbf{R}^d \longrightarrow \mathbf{R}$,

$$\forall\, x\,, \quad g(x)=b+w_1x_1+\ldots+w_dx_d\,, \tag{2}$$

where $x=(x_1,\ldots,x_d)\in\mathbf{R}^d$, such that for any $(x_i,y_i)\in\mathcal{S}$, $y_i g(x_i)>0$.

Denoting by $w=(w_1,\ldots,w_d)^T$ the vector whose entries are the coefficients of g, we say that $\mathcal{S}$ is separated without errors by the hyperplane

$$H_{w,b}\ :\quad w^Tx+b=0\,. \tag{3}$$

Obviously all examples coming from the class of label 1 belong to the positive semi-space, and all examples coming from the class of label -1 belong to the negative semi-space defined by $H_{w,b}$. For this reason, $H_{w,b}$ is called a solution of the separating problem.

In a SVM-based approach, the search for a solution $H_{w,b}$ is usually formulated as a constraint optimization problem on the objective function $\Phi(w)=\dfrac{1}{2}\|w\|^2$,

$$\begin{cases} \min \Phi(w) \\ y_i\left(w^Tx_i+b\right)\geq 1\,, \quad i=\overline{1,N}\,. \end{cases} \tag{4}$$

If w^* is a solution of (4), then H_{w^*,b^*} is called an optimal separating hyperplane, where the computation of w^* and b^* is carried out using the following algorithm

Algorithm $SVM1$ ([9])

`Input:` $\mathcal{S}=\left\{ (x_i,y_i) \mid x_i\in\mathbf{R}^d,\ y_i\in\{-1,1\},\ i=\overline{1,N}\right\}$

Step 1. `Compute the matrix` $D=(d_{ik})$ `of entries,` $d_{ik}=y_iy_k\,(x_i)^T\,x_k$, $i,k=\overline{1,N}$ `;`

Step 2. `Solve the constrained optimization problem`

$$\begin{cases} \alpha^*=\texttt{arg}\left(\max\limits_{\alpha\in\mathbf{R}^N}\left(\alpha^T\mathbf{1}-\dfrac{1}{2}\,\alpha^TD\alpha\right)\right), \\ \alpha_i\geq 0\,, \quad \forall\, 1\leq i\leq N\,, \\ \sum\limits_{i=1}^{N}\alpha_iy_i=0\,, \end{cases} \tag{5}$$

`If` $\alpha_i^*>0$ `then` x_i `is called the support vector.`

Step 3. `Select two support vectors` x_r, x_s `such that` $\alpha_r^*>0$, $\alpha_s^*>0$, $y_r=-1$, $y_s=1$.

Step 4. `Compute the parameters` w^*,b^* `of the optimal separating hyperplane, and the width of the separating area` $\rho(w^*,b^*)$,

$$\begin{cases} w^*=\sum\limits_{i=1}^{N}\alpha_i^*y_ix_i\,, \\ b^*=-\dfrac{1}{2}\,(w^*)^T\,(x_r+x_s)\,, \\ \rho(w^*,b^*)=\dfrac{2}{\|w^*\|} \end{cases} \tag{6}$$

`Output:` $w^*,b^*,\rho(w^*,b^*)$.

A linear separable sample is represented in figure 1a. The straight lines d_1, d_2, d_3 and d_4 are solutions for the separating problem of $\mathcal{S}$, d_4 corresponds to the optimal separating hyperplane. The examples placed at the minimum distance to the optimum separating hyperplane are the support vectors.

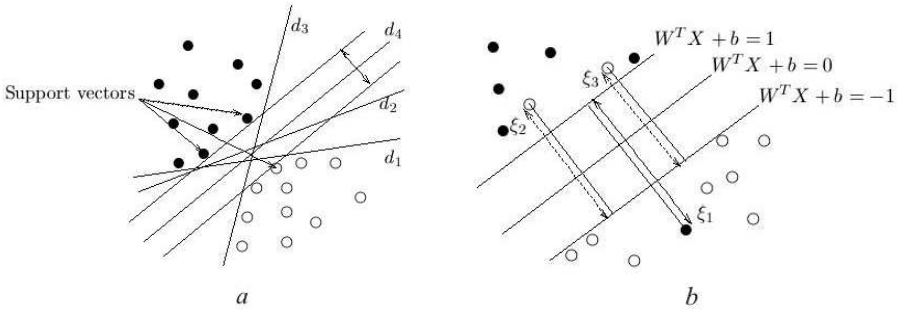

Figure 1: *a) Optimal separating hyperplane; b) Classification errors.*

In case of non-linearly separable samples the idea is to determine a separating hyperplane such that the number of misclassified examples is minimized.

The problem of designing the optimal hyperplane in case of non-linearly separable samples has been approached several ways. The approach introduced by Cortes and Vapnik ([3]) uses the error function

$$\Phi_\sigma(\xi) = \sum_{i=1}^{N} \xi_i^\sigma \,, \tag{7}$$

where the slack variables $\xi_i \,, 1 \leq i \leq N$, are taken as indicators for the classification errors (see figure 1*b*), and σ is a positive real number.

The optimality is expressed in terms of the objective function $\Phi : \mathbf{R}^d \times \mathbf{R}^N \longrightarrow [0, +\infty)$

$$\Phi(w, \xi) = \frac{1}{2} \|w\|^2 + c\, F\left(\sum_{i=1}^{N} \xi_i^\sigma\right) = \frac{1}{2} \sum_{j=1}^{n} w_j^2 + c\, F\left(\sum_{i=1}^{N} \xi_i^\sigma\right) \,, \tag{8}$$

where c is a given positive constant, $\xi = (\xi_1, \ldots, \xi_N)$, and F is a monotone convex function, $F(0) = 0$.

The idea is to compute a subset of $\mathcal{S}$, say $\{(x_{i_1}, y_{i_1}), \ldots, (x_{i_k}, y_{i_k})\}$, by minimizing $\Phi_\sigma(\xi)$, such that there exists an optimal hyperplane for $\mathcal{S} \setminus \{(x_{i_1}, y_{i_1}), \ldots, (y_{i_k}, y_{i_k})\}$. This optimal hyperplane is referred as the soft margin hyperplane ([3]).

The soft margin hyperplane is a solution of the constrained optimization problem

$$\begin{cases} \arg\left(\min\limits_{w \in \mathbf{R}^d} (\Phi(w, \xi))\right) \\ \\ y_i \left(w^T x_i + b\right) \geq 1 - \xi_i \,, \; \forall\, 1 \leq i \leq N \,, \\ \xi_i \geq 0 \,, \; \forall\, 1 \leq i \leq N \,, \end{cases} \tag{9}$$

The samples represented in figure 1*b*, correspond to the non-linearly separable case. The soft margin hyperplane, the separating area, and slack variables are indicated in figure 1*b*.

The computation of soft margin hyperplane is carried out by the following algorithm.

Algorithm $SVM2$ ([9])

`Input:` $\mathcal{S} = \left\{ (x_i, y_i) \mid x_i \in \mathbf{R}^n \,,\; y_i \in \{-1, 1\} \,,\; i = \overline{1, N} \right\}, \quad c \in (0, \infty)$;

Step 1. `Compute the matrix` $D = (d_{ik})$ `of entries,` $d_{ik} = y_i y_k \left(x_i\right)^T x_k \,,\; i, k = \overline{1, N}$;

Step 2. `Solve the constrained optimization problem`

$$\begin{cases} \alpha^* = \mathtt{arg}\left(\max\limits_{\alpha \in \mathbf{R}^N} \left(\alpha^T \mathbf{1} - \frac{1}{2}\, \alpha^T D \alpha - \frac{\left(\alpha_{max}\right)^2}{4\, c}\right)\right) \,, \\ \alpha_i \geq 0 \,, \quad \forall\, 1 \leq i \leq N \,, \\ \sum\limits_{i=1}^{N} \alpha_i Y_i = 0 \,, \end{cases} \tag{10}$$

where $\alpha_{max} = \max\{\alpha_1, \ldots, \alpha_N\}$

Step 3. Select two support vectors x_r, x_s such that $\alpha_r^* > 0$, $\alpha_s^* > 0$, $y_r = -1$, $y_s = 1$.

Step 4. Compute the parameters w^*, b^* of the soft margin hyperplane, and the width of the separating area $\rho(w^*, b^*)$, according to (6).

Output: $w^*, b^*, \rho(w^*, b^*)$.

3 Unsupervised learning (clustering) using the k-means method

Center-based clustering algorithms are very efficient for clustering large databases and high-dimensional databases. They have own objective functions which define how good a clustering solution is, the goal being to minimize the objective function. Clusters found by center-based algorithms have convex shapes and each cluster is represented by a center. The k-means algorithm introduced by MacQueen ([8]) was designed to cluster numerical data, each cluster having a center called the *mean*.

Let $\mathcal{D} = \{x_1, \ldots, x_N\} \subset \mathbf{R}^d$ be the data set, k a given positive integer, and $\mathcal{C}_1, \ldots, \mathcal{C}_k$ pairwise disjoint clusters of $\mathcal{D}$, that is, $\bigcup_{i=1}^{k} \mathcal{C}_i = \mathcal{D}$, $\mathcal{C}_i \cap \mathcal{C}_j$, $\forall\, i \neq j$. If we denote by $\mu(\mathcal{C}_i)$ the center of $\mathcal{C}_i$ then *the inertia momentum (error)* is expressed by

$$\varepsilon = \sum_{i=1}^{k} \sum_{x \in \mathcal{C}_i} d^2(x, \mu(\mathcal{C}_i)) \,, \tag{11}$$

where d is a convenable distance function on $\mathbf{R}^d$. In the following we take d as being the Euclidean distance on $\mathbf{R}^d$, $d(x, y) = \|x - y\|$.

The k-means methods proceeds, for a given initial k clusters, by allocating the remaining data to the nearest clusters and then repeatedly changing the membership of the clusters according to the error function until the error function does not change significantly or the membership of the clusters no longer changes.

The k-means algorithm can be treated as an optimization problem where the goal is to minimize a given objective function under certain constrains.

We denote by $\mathcal{C}$ the set of all subsets of $\mathbf{R}^d$ of cardinal k; any particular $Q = \{q_1, \ldots, q_k\} \in \mathcal{C}$ is called a set of possible centers.

A system of k pairwise disjoint clusters of $\mathcal{D}$ can be obviously represented in terms a matrix $W = (w_{il}) \in \mathcal{M}_{N \times k}(\mathbf{R})$ such that

$$\begin{aligned} &(i) \quad w_{il} \in \{0, 1\}\,, \quad i = \overline{1, N}\,,\ l = \overline{1, k} \\ &(ii) \quad \sum_{l=1}^{k} w_{il} = 1\,, \quad i = \overline{1, N}\,. \end{aligned} \tag{12}$$

The k-means algorithm can be formulated as the constrained optimization problem:

$$\begin{cases} \min\limits_{W \in \mathcal{M}_{N \times k}(\mathbf{R}),\ Q \in \mathcal{C}} P(W, Q) \\ w_{il} \in \{0, 1\}\,, \quad i = \overline{1, N}\,,\ l = \overline{1, k}\,, \\ \sum\limits_{l=1}^{k} w_{il} = 1\,, \quad i = \overline{1, N}\,, \end{cases} \tag{13}$$

where the objective function is defined as

$$P(W, Q) = \sum_{i=1}^{N} \sum_{l=1}^{k} w_{il} \|x_i - q_l\|^2 \,. \tag{14}$$

The problem (13) can be solved by decomposing it into two simpler problems P_1 and P_2, and then iteratively solving them, where

P_1. Fix $Q = \widehat{Q} \in \mathcal{C}$ and solve the reduced constrained optimization problem for $P\left(W, \widehat{Q}\right)$.

P_2. Fix $W = \widehat{W} \in \mathcal{M}_{N\times k}(\mathbf{R})$ and solve the reduced unconstrained optimization problem for $P\left(\widehat{W}, Q\right)$.

The solutions of these problems can be derived by straightforward computations, and they are given by the following theorems:

Theorem 1 *For any fixed $\widehat{Q} = \{\widehat{q}_1, \ldots, \widehat{q}_k\}$ a set of centers, the function $P\left(W, \widehat{Q}\right)$ is minimized if and only if W satisfies the conditions*

$$\begin{aligned} &w_{il} = 0 \Longleftrightarrow \|x_i - \widehat{q}_l\| > \min_{1\le t\le k} \|x_i - \widehat{q}_t\| \ , \\ &w_{il} = 1 \Longrightarrow \|x_i - \widehat{q}_l\| = \min_{1\le t\le k} \|x_i - \widehat{q}_t\| \ , \\ &\sum_{j=1}^{k} w_{ij} = 1 \, , \ \textit{for any} \ \ i = \overline{1,N} \, , \, l = \overline{1,k}. \end{aligned}$$

Note that in general, for any given $\widehat{Q}$ there are more solutions of $W^{(0)}$ type because any particular data x_i can be at minimum distance to more than one center of $\widehat{Q}$.

Theorem 2 *For any fixed $\widehat{W}$ satisfying the constrains of (13), the function $P\left(\widehat{W}, Q\right)$ is minimized if and only if*

$$q_l = \frac{\sum_{i=1}^{N} \widehat{w}_{il} x_i}{\sum_{i=1}^{N} \widehat{w}_{il}} \, , \quad l = \overline{1,k}.$$

The k-means algorithm viewed as an optimization process for solving (13) is as follows

The algorithm k-MOP

```
Input:  D - the data set,
        k - the pre-specified number of clusters,
        d - the data dimensionality,
        T - threshold on the maximum number of iterations.
Initializations:  Q^(0), t ⟵ 0
Solve P(W, Q^(0)) and get W^(0)
sw ⟵ false
repeat
        Ŵ ⟵ W^(t)
        solve P(Ŵ, Q) and get Q^(t+1)
        if P(Ŵ, Q^(t)) = P(Ŵ, Q^(t+1)) then
              sw ⟵ true
              output (Ŵ, Q^(t+1))
        else
              Q̂ ⟵ Q^(t+1)
              solve P(W^(t), Q̂) and get W^(t+1)
              if P(W^(t), Q̂) = P(W^(t+1), Q̂) then
                    sw ⟵ true
                    output (W^(t+1), Q̂, )
              endif
```

endif
$t \longleftarrow t+1$
until sw or $t > T$.

Note that the computational complexity of the algorithm k-**MOP** is $\mathcal{O}(Nkd)$ per iteration. The sequence of values $P\left(W^{(t)}, Q^{(t)}\right)$ where $W^{(t)}$, $Q^{(t)}$ are computed by k-**MOP** is strictly decreasing, therefore the algorithm converges to a local minimum of the objective function.

4 The combined separating technique based on SVM and the k-means algorithm

At first sight, it seems unreasonable to compare a supervised technique to an unsupervised one, mainly because they refer to totally different situations. On one hand the supervised techniques are applied in case the data set consists of correctly labeled objects, and on the other hand the unsupervised methods deal with unlabeled objects. However our point is to combine SVM and k-means algorithm, in order to obtain a new design of a linear classifier.

The aim of the experimental analysis is to evaluate the performance of the linear classifier resulted from the combination of the supervised SVM method and the 2-means algorithm.

Our method can be applied to whatever data, either linear separable or non-linear separable. Obviously in case of non-linear separable data the classification can not be performed without errors and in this case the number of misclassified examples is the most reasonable criterion for performance evaluation. Of a particular importance is the case of linear separable data, in this case the performance being evaluated in terms of both, misclassified examples and the generalization capacity expressed in terms of the width of separating area. In real live situations, usually is very difficult or even impossible to established whether the data represents a linear/non-linear separable set. In using the $SVM1$ approach we can identify which case the given data set belongs to. For linear separable data, $SVM1$ computes a separation hyperplane optimal from the point of view of the generalization capacity. In case of a non-linear separable data $SVM2$ computes a linear classifier that minimizes the number of misclassified examples. A series of developments are based on non-linear transforms represented be kernel functions whose range are high dimensional spaces. The increase of dimensionality and the convenable choice of the kernel aim to transform a non-linear separable problem into a linear separable one. The computation complexity corresponding to kernel-based approaches is significantly large therefore in case the performance of the algorithm $SVM1$ proves reasonable good it could be taken as an alternative approach of a kernel-based SVM. We perform a comparative analysis on data consisting of examples generated from two dimensional Gaussian distributions.

In case of a non-linear separable data set, using the k-means algorithm, we get a system of pairwise disjoint clusters together with the set of their centers representing a local minimum point of the criterion (13), the clusters being linear separable when $k = 2$. Consequently, the $SVM1$ algorithm computes a linear classifier that separates without errors the resulted clusters.

Our procedure is described as follows

Input: $\mathcal{S} = \left\{ (x_i, y_i) \mid x_i \in \mathbf{R}^n,\ y_i \in \{-1, 1\},\ i = \overline{1, N} \right\}$

Step 1. Compute the matrix $D = (d_{ik})$ of entries, $d_{ik} = y_i y_k \left(x_i\right)^T x_k,\ i, k = \overline{1, N}$;
$sh \longleftarrow true$

Step 2. If the constrained optimization problem (5) does not have solution then
$sh \longleftarrow false$
input $c \in (0, \infty)$, for hyperplane soft margin
Solve the constrained optimization problem (10)
endif

Step 3. Select x_r, x_s such that $\alpha_r^* > 0$, $\alpha_s^* > 0$, $y_r = -1$, $y_s = 1$;
Compute the parameters w^*, b^* of the separating hyperplane,

and the width of the separating area, $\rho(w^*, b^*)$ according to (6);
Compute the width of the separating area, $\rho(w^*, b^*) = \frac{2}{\|w^*\|}$;

Step 4. if *not sh* then
compute nr_err1 - the numbers of examples incorrect classified
compute $err1$ - error classification
endif

Step 5. The set $\mathcal{D} = \{x_i \mid x_i \in \mathbf{R}^d,\ i = \overline{1,N}\}$ is divided in two clusters $\mathcal{C}_1$ and $\mathcal{C}_2$ using 2-means, marked out with $y_i' = 1$ and $y_i' = -1$ respectively.

Step 6. Apply *algorithm SVM1* for
$\mathcal{S}' = \{(x_i, y_i') \mid x_i \in \mathbf{R}^d,\ y_i' \in \{-1, 1\},\ i = \overline{1,N}\}$
and obtain the parameters for optimal separating hyperplane: $w_1^*, b_1^*, \rho(w_1^*, b_1^*)$
compute nr_err2 - the numbers of examples incorrect classified by $2-means$
compute $err2$ - error classification after $2-means$

Output: $w^*, b^*, \rho(w^*, b^*)$, nr_err1, $err1$, $w_1^*, b_1^*, \rho(w_1^*, b_1^*)$, nr_err2, $err2$.

5 Comparative analysis and experimental results

The experimental analysis is based on a long series of tests performed on linear/non-linear separable simulated data of different volumes. The analysis aims to derive conclusions concerning:

1. The statistical properties (the empirical means, covariance matrices, eigenvalues, eigenvectors) of the clusters computed by the 2-means algorithm as compared to their counterparts corresponding to the true distributions they come from.
2. The comparison of the performances corresponding to the linear classifier resulted as a combination of SVM and the 2-means algorithm described in section 4 and $SVM2$ in terms of the empirical error.
3. The analysis concerning the influences of the samples sizes on the performance of the procedure described in section 4.
4. The quality of cluster characterization in terms of the principal directions given by a system of unit orthogonal eigenvectors of the sample covariance and empirical covariance matrices of the computed clusters. The analysis aimed to derive conclusions concerning the contributions of each principal direction, and for this reason, some tests were performed on data whose first principal component is strongly dominant, and when the principal directions are of the same importance respectively.

The tests were performed on data generated from normal two-dimensional distributions $\mathcal{N}(\mu_i, \Sigma_i)$, $i = 1, 2$ of volumes N_1 and N_2. The sample covariance matrices are denoted by $\widehat{\mu}_i, \widehat{\Sigma}_i$, $i = 1, 2$. The centers and the empirical covariance matrices corresponding to the clusters computed by the 2-means algorithm are denoted by $\overline{\mu}_i, \overline{\Sigma}_i$, $i = 1, 2$. We denote by Z_i , $\widehat{Z}_i$, $\overline{Z}_i$, $i = 1, 2$ orthogonal matrices having as columns unit eigenvector of Σ_i , $\widehat{\Sigma}_i$, $\overline{\Sigma}_i$, $i = 1, 2$ respectively.

Test 1:
$N_1 = N_2 = 50$, $\mu_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, $\Sigma_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0.25 \end{pmatrix}$, $\mu_2 = \begin{pmatrix} 2 \\ 3 \end{pmatrix}$, $\Sigma_2 = \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix}$.
The matrices Z_1 , Z_2 and their eigenvalues are

$\lambda_1^{(1)} = 0.25$, $\lambda_2^{(1)} = 1$, $Z_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$, $\lambda_1^{(2)} = 0.5$, $\lambda_2^{(2)} = 0.5$, $Z_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$.

The set is non-linear separable and it is represented in figure $2i)a$. In this case we get

$\widehat{\mu}_1 = \begin{pmatrix} 0.92 \\ 1.00 \end{pmatrix}$, $\widehat{\Sigma}_1 = \begin{pmatrix} 0.85 & 0.086 \\ 0.08 & 0.25 \end{pmatrix}$, $\widehat{\mu}_2 = \begin{pmatrix} 1.98 \\ 2.87 \end{pmatrix}$, $\widehat{\Sigma}_2 = \begin{pmatrix} 0.44 & 0.09 \\ 0.09 & 0.63 \end{pmatrix}$.

the matrices $\widehat{Z}_1$, $\widehat{Z}_2$ and their eigenvalues being

$\widehat{\lambda}_1^{(1)}=0.24\,,\quad \widehat{\lambda}_2^{(1)}=0.86,\ \widehat{Z}_1=\begin{pmatrix} 0.14 & -0.98 \\ -0.98 & -0.14 \end{pmatrix},\quad \widehat{\lambda}_1^{(2)}=0.40\,,\quad \widehat{\lambda}_2^{(2)}=0.67,\ \widehat{Z}_2=\begin{pmatrix} -0.92 & 0.38 \\ 0.38 & 0.92 \end{pmatrix}.$

Using the $SVM2$ with $c = 70$ we get the classification error $class\_error = 14.70$, the number of misclassified samples $n\_errors = 13$ and the width of separating area is $\rho = 0.61$. The value of the error coefficient defined as the ratio of the number of misclassified samples and total volume of the data is $c\_error = 0.13\%$. The soft margin line d_1 is represented in figure $2i)b$.

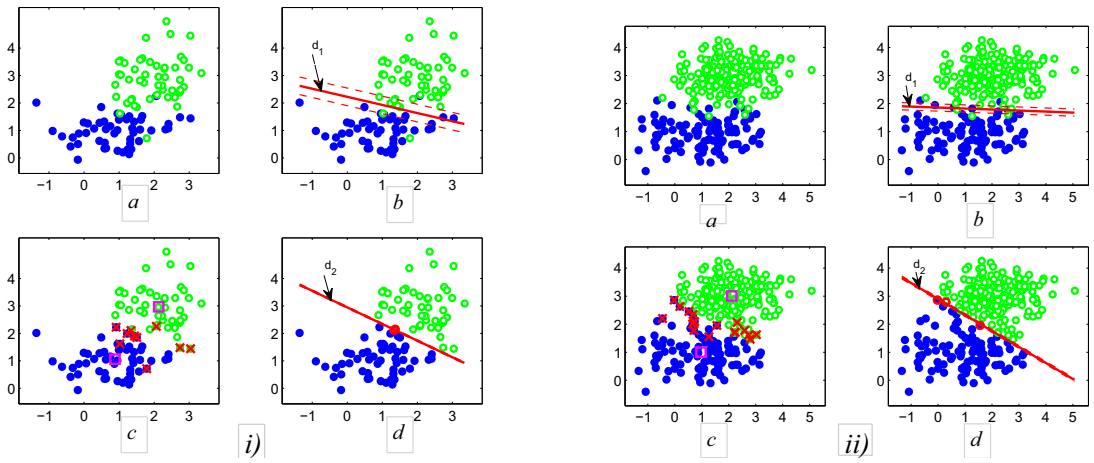

Figure 2: *i) The classification of the data set in test 1; ii) The classification of the data set in test 2.*

By applying the 2-means algorithm we get clusters whose empirical means and covariances are

$\overline{\mu}_1=\begin{pmatrix} 0.88 \\ 1.06 \end{pmatrix},\quad \overline{\Sigma}_1=\begin{pmatrix} 0.64 & 0.05 \\ 0.05 & 0.30 \end{pmatrix},\quad \overline{\mu}_2=\begin{pmatrix} 2.13 \\ 2.96 \end{pmatrix},\quad \overline{\Sigma}_2=\begin{pmatrix} 0.41 & -0.06 \\ -0.06 & 0.56 \end{pmatrix}.$

The matrices $\overline{Z}_1$, $\overline{Z}_2$ and their eigenvalues are

$\overline{\lambda}_1^{(1)}=0.29\,,\quad \overline{\lambda}_2^{(1)}=0.65,\ \overline{Z}_1=\begin{pmatrix} 0.14 & -0.98 \\ -0.98 & -0.14 \end{pmatrix},\quad \overline{\lambda}_1^{(2)}=0.39\,,\ \overline{\lambda}_2^{(2)}=0.58,\ \overline{Z}_2=\begin{pmatrix} -0.92 & -0.37 \\ -0.37 & 0.92 \end{pmatrix},$

the number of misclassified samples is 10 and the clusters are represented in figure $2i)c$.

Note that the computed centers and clusters are not influenced by the initial centers. In figure $2i)c$ are represented the clusters computed by the 2-means algorithm for randomly selected initial centers. The separating line d_2 resulted by applying the $SVM1$ algorithm to the data represented by the clusters computed by the 2-means algorithm is represented in figure $2i)d$.

Test 2:

$N_1=100\,,\quad N_2=200,\quad \mu_1=\begin{pmatrix} 1 \\ 1 \end{pmatrix},\quad \Sigma_1=\begin{pmatrix} 1 & 0 \\ 0 & 0.25 \end{pmatrix},\quad \mu_2=\begin{pmatrix} 2 \\ 3 \end{pmatrix},\quad \Sigma_2=\begin{pmatrix} 1 & 0 \\ 0 & 0.25 \end{pmatrix},$

$\lambda_1^{(1)}=0.25\,,\quad \lambda_2^{(1)}=1,\quad Z_1=\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},\quad \lambda_1^{(2)}=0.25\,,\quad \lambda_2^{(2)}=1,\quad Z_2=\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$

$\widehat{\mu}_1=\begin{pmatrix} 1.12 \\ 0.92 \end{pmatrix},\quad \widehat{\Sigma}_1=\begin{pmatrix} 1.35 & 0.04 \\ 0.04 & 0.26 \end{pmatrix},\quad \widehat{\mu}_2=\begin{pmatrix} 2.01 \\ 3.00 \end{pmatrix},\quad \widehat{\Sigma}_2=\begin{pmatrix} 0.86 & 0.05 \\ 0.05 & 0.25 \end{pmatrix},$

$\widehat{\lambda}_1^{(1)}=0.26\,,\ \widehat{\lambda}_2^{(1)}=1.35,\ \widehat{Z}_1=\begin{pmatrix} 0.03 & -0.99 \\ -0.99 & -0.03 \end{pmatrix},\quad \widehat{\lambda}_1^{(2)}=0.25\,,\ \widehat{\lambda}_2^{(2)}=0.87,\ \widehat{Z}_2=\begin{pmatrix} 0.09 & -0.99 \\ -0.99 & -0.09 \end{pmatrix}.$

The data set is non-linear separable and it is represented in figure $2ii)a$. Applying the $SVM2$ for $c = 5$ we obtain the soft margin line d_1 represented in figure $2ii)b$ and $class\_error = 19.12$, $n\_errors = 13$, $\rho = 0.25$, $c\_error = 0.043\%$.

The clusters computed by the 2-means algorithm are represented in figure $2ii)c$ and their statistical characteristics are

$$\overline{\mu}_1=\begin{pmatrix}0.96\\1.00\end{pmatrix},\quad \overline{\Sigma}_1=\begin{pmatrix}1.19 & -0.10\\-0.10 & 0.38\end{pmatrix},\quad \overline{\mu}_2=\begin{pmatrix}2.10\\3.00\end{pmatrix},\quad \overline{\Sigma}_2=\begin{pmatrix}0.76 & -0.02\\-0.02 & 0.28\end{pmatrix},$$

$$\overline{\lambda}_1^{(1)}=0.37,\ \overline{\lambda}_2^{(1)}=1.20,\ \overline{Z}_1=\begin{pmatrix}-0.12 & -0.99\\-0.99 & 0.12\end{pmatrix},\quad \overline{\lambda}_1^{(2)}=0.27,\ \overline{\lambda}_2^{(2)}=0.76,\ \overline{Z}_2=\begin{pmatrix}-0.05 & -0.99\\-0.99 & 0.05\end{pmatrix}.$$

In this case the number of misclassified samples is 18. Note that the initial choice of the centers does not influence significantly the computed centers and clusters. For instance in figure $2ii)c$ are represented the resulted clusters in case of randomly selected initial centers.

The separating line d_2 computed by the algorithm $SVM1$ applied to the data represented by these clusters is represented in figure $2ii)d$.

Test 3:

$$N_1=N_2=50,\quad \mu_1=\begin{pmatrix}1\\1\end{pmatrix},\quad \Sigma_1=\begin{pmatrix}1 & 0\\0 & 0.25\end{pmatrix},\quad \mu_2=\begin{pmatrix}3\\4\end{pmatrix},\quad \Sigma_2=\begin{pmatrix}0.5 & 0\\0 & 0.5\end{pmatrix},$$

$$\lambda_1^{(1)}=0.25,\quad \lambda_2^{(1)}=1,\quad Z_1=\begin{pmatrix}0 & 1\\1 & 0\end{pmatrix},\quad \lambda_1^{(2)}=0.5,\quad \lambda_2^{(2)}=0.5,\quad Z_2=\begin{pmatrix}1 & 0\\0 & 1\end{pmatrix},$$

$$\widehat{\mu}_1=\begin{pmatrix}0.76\\1.00\end{pmatrix},\quad \widehat{\Sigma}_1=\begin{pmatrix}1.17 & -0.06\\-0.06 & 0.21\end{pmatrix},\quad \widehat{\mu}_2=\begin{pmatrix}2.87\\4.03\end{pmatrix},\quad \widehat{\Sigma}_2=\begin{pmatrix}0.56 & 0.00\\0.00 & 0.31\end{pmatrix},$$

$$\widehat{\lambda}_1^{(1)}=0.21,\ \widehat{\lambda}_2^{(1)}=1.18,\ \widehat{Z}_1=\begin{pmatrix}-0.07 & -0.99\\-0.99 & 0.07\end{pmatrix},\quad \widehat{\lambda}_1^{(2)}=0.31,\ \widehat{\lambda}_2^{(2)}=0.56,\ \widehat{Z}_2=\begin{pmatrix}0.03 & -0.99\\-0.99 & -0.03\end{pmatrix}.$$

The data set is linear separable and it is represented in figure $3i)a$. Applying the $SVM1$ we obtain the soft margin line d_1 represented in $3i)b$ and $\rho=1.19$.

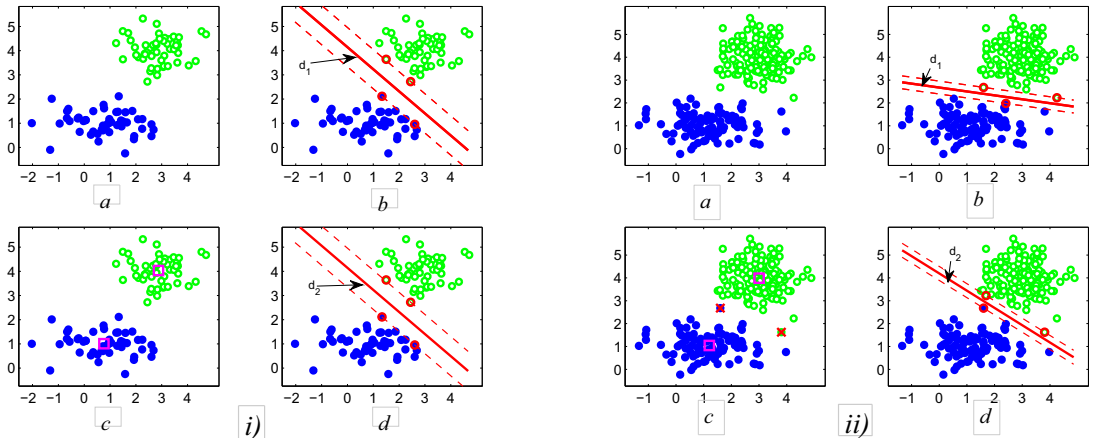

Figure 3: *i) The classification of the data set in test 3; ii) The classification of the data set in test 4.*

The clusters computed by the 2-means algorithm are represented in figure $3i)c$ and they are the same as in initial data set whatever the initial choice of the centers is. So, the statistical characteristics are

$$\overline{\mu}_1=\widehat{\mu}_1,\ \overline{\Sigma}_1=\widehat{\Sigma}_1,\ \overline{\mu}_2=\widehat{\mu}_2,\ ,\ \overline{\Sigma}_2=\widehat{\Sigma}_2,$$

$$\overline{\lambda}_1^{(1)}=\widehat{\lambda}_1^{(1)},\ \overline{\lambda}_2^{(1)}=\widehat{\lambda}_2^{(1)},\ \overline{Z}_1=\widehat{Z}_1,\ \overline{\lambda}_1^{(2)}=\widehat{\lambda}_1^{(2)},\ \overline{\lambda}_2^{(2)}=\widehat{\lambda}_2^{(2)},\ \overline{Z}_2=\widehat{Z}_2,$$

and the separating line d_2 computed by the algorithm $SVM1$ and represented in figure $3i)d$ coincides with d_1.

Test 4:

$N_1 = 100\,, \quad N_2 = 150, \quad \mu_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \quad \Sigma_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0.25 \end{pmatrix}, \quad \mu_2 = \begin{pmatrix} 3 \\ 4 \end{pmatrix}, \quad \Sigma_2 = \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix}.$

$\lambda_1^{(1)} = 0.25\,, \quad \lambda_2^{(1)} = 1, \quad Z_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \lambda_1^{(2)} = 0.5\,, \quad \lambda_2^{(2)} = 0.5, \quad Z_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$

$\widehat{\mu}_1 = \begin{pmatrix} 1.22 \\ 1.03 \end{pmatrix}, \quad \widehat{\Sigma}_1 = \begin{pmatrix} 1.04 & -0.03 \\ -0.03 & 0.24 \end{pmatrix}, \quad \widehat{\mu}_2 = \begin{pmatrix} 2.98 \\ 3.99 \end{pmatrix}, \quad \widehat{\Sigma}_2 = \begin{pmatrix} 0.48 & -0.01 \\ -0.01 & 0.43 \end{pmatrix}.$

$\widehat{\lambda}_1^{(1)} = 0.24\,, \ \widehat{\lambda}_2^{(1)} = 1.04, \ \widehat{Z}_1 = \begin{pmatrix} -0.04 & -0.99 \\ -0.99 & 0.04 \end{pmatrix}, \quad \widehat{\lambda}_1^{(2)} = 0.42\,, \ \widehat{\lambda}_2^{(2)} = 0.49, \ \widehat{Z}_2 = \begin{pmatrix} -0.27 & -0.9 \\ -0.96 & 0.27 \end{pmatrix}.$

The data set is linear separable and it is represented in figure $3ii)a$. Applying the $SVM1$ we obtain the soft margin line d_1 represented in $3ii)b$ and $\rho = 0.55$.

The clusters computed by the 2-means algorithm are represented in figure $3ii)c$ and their statistical characteristics are

$\overline{\mu}_1 = \begin{pmatrix} 1.20 \\ 1.04 \end{pmatrix}, \quad \overline{\Sigma}_1 = \begin{pmatrix} 0.98 & -0.04 \\ -0.04 & 0.26 \end{pmatrix}, \quad \overline{\mu}_2 = \begin{pmatrix} 3.00 \\ 3.98 \end{pmatrix}, \quad \overline{\Sigma}_2 = \begin{pmatrix} 0.48 & -0.04 \\ -0.04 & 0.45 \end{pmatrix},$

$\overline{\lambda}_1^{(1)} = 0.26\,, \ \overline{\lambda}_2^{(1)} = 0.98, \ \overline{Z}_1 = \begin{pmatrix} -0.05 & -0.99 \\ -0.99 & 0.05 \end{pmatrix}, \quad \overline{\lambda}_1^{(2)} = 0.42\,, \ \overline{\lambda}_2^{(2)} = 0.51, \ \overline{Z}_2 = \begin{pmatrix} -0.60 & -0.79 \\ -0.79 & 0.60 \end{pmatrix}.$

In this case the number of misclassified samples is 2 and the initial centers are randomly selected. The separating line d_2 computed by the algorithm $SVM1$ applied to the data represented by these clusters is represented in figure $3ii)d$.

References

[1] Abe, S., *Support Vector Machines for Pattern Classification*, Springer-Verlag, 2005.

[2] Burges, C.J.C., *A Tutorial on Support Vector Machines for Pattern Recognition*, Data Mining and Knowledge Discovery, **2** (1998), 121-167.

[3] Cortes, C., Vapnik, V., *Support-vector networks*, Machine Learning, 20(3), 273-297, 1995.

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

[5] Gan, G., Ma C., Wu, J., *Data Clustering: Theory, Algorithms and Applications*, SIAM, 2007.

[6] Gunn, S.R., *Support Vector Machines for Classification and Regression*, University of Southampton, Technical Report, 1998.

[7] State, L., Paraschiv-Munteanu, I., *Introducere in teoria statistică a recunoaşterii formelor*, Editura Universitaţii din Piteşti, 2009.

[8] MacQueen, J.B., *Some Methods for classification and Analysis of Multivariate Observations*, Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 281-297, 1967.

[9] Paraschiv-Munteanu, I., *Support Vector Machine in solving pattern recognition tasks*, Scientific Bulletin, No.16/2009, University of Pitesti, ISSN 1453-116x.

[10] Vapnik, V.N., *The Nature of Statistical Learning Theory*, New York, Springer Verlag, 1995.

[11] Vapnik, V.N., *Statistical Learning Theory*, New York, Wiley-Interscience, 1998.

[12] Xu, R., Wunsch, D.C.II, *Clustering*, Wiley&Sons, 2009.

Luminiţa State
University of Piteşti
Faculty of Mathematics and Computer Science
1 Târgu din Vale St., Piteşti 110040
ROMANIA
E-mail: *lstate@clicknet.ro*

Iuliana Paraschiv-Munteanu
University of Bucharest
Faculty of Mathematics and Computer Science
14 Academiei St., Bucharest 010014
ROMANIA
E-mail: *pmiulia@fmi.unibuc.ro*