

A Maximal Margin Hypersphere SVM

Rui Malha[1] and Paula Amaral[1,2(✉)]

[1] Nova SST—FCT Nova, Campus de Caparica, 2829-516 Caparica, Portugal
paca@fct.unl.pt
[2] NovaMaths CMA Nova, Campus de Caparica, 2829-516 Caparica, Portugal

Abstract. In this work we propose a generalization of the Support Vector Machine (SVM) method in which the separator is a curve, but the concept of margin and maximization of the margin is still present. The idea of using different functions for the separation has been explored in particular in the scope of hyperspheres. However, most of these proposals use two spheres, using concepts different from maximal margin or one sphere but with a poor performance when data from the classes have a linear shape. In this paper we present a formulation of the linear SVM that generalizes it to a spherical separation shape, but still maximizing the margin. A linear relaxation of this quadratic formulation is also presented. The performance of these two formulations for classification purpose is tested and the results are encouraging.

Keywords: SVM · Automatic classification · Non-linear SVM

1 Introduction

Today we have sensors connected to houses, bodies, cars, machines, trees, animals. The act of monitoring reached a hyperbolic state in the era of Big Data, 4th industrial revolution, internet of things (IOT), smart cities, smart cars, smart appliances.

Interpreting and classifying these types of data requires good classification methods. Support Vector Machine (SVM) have been used with success in classification [2] and several generalizations of the boundary function have been proposed in the literature.

In this paper we propose a spherical separation, maintaining the concept of maximal margin. The key feature consists in letting the radius out of the minimization purpose so that, the sphere can approximate a linear separation by enlarging the radius and putting the center away from the centroid of the points. Most methods seek to minimize the ratio and so they have a poor performance for data with linear shape. We formulate and solve this nonlinear SVM for a set of generated data and for set of real data from a repository commonly used for this type of studies. This leads to a quadratic optimization problem and we

This work is funded by national funds through the FCT - Fundação para a Ciência e a Tecnologia, I.P., under the scope of the project UIDB/00297/2020 (Center for Mathematics and Applications).

© Springer Nature Switzerland AG 2021
O. Gervasi et al. (Eds.): ICCSA 2021, LNCS 12953, pp. 304–319, 2021.
https://doi.org/10.1007/978-3-030-86976-2_21

also propose a linear formulation that can give an approximate solution to this quadratic problem.

The paper is organized in the following way: After this introduction we present the classical SVM method introducing the formulation from a pure optimization perspective without resorting to classical formulas of distance of a point to a line. This helps to understand the formulation of the classical SVM. In Sect. 3 we present a survey in spherical SVM to highlight the contribution of the methodology proposed in this paper. In Sect. 4 we develop our method including the linear formulation and, present in Sect. 5, before the conclusions, the numerical experience.

2 The Standard SVM

In this section we are going to review the SVM method and describe the fundamentals of the theory behind, using a pure optimization interpretation.

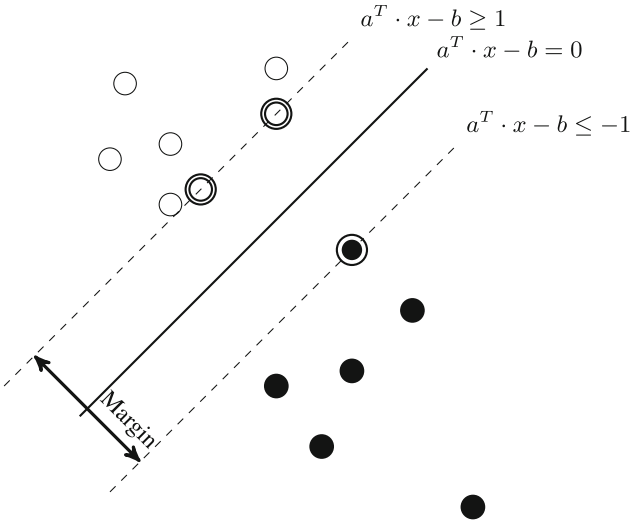


Fig. 1. SVM with two features ($\mathbb{R}^2$)

SVM is a supervised classification method. It uses two sets of labeled data, one to find a frontier that separates two or more classes of data (the training examples) and the other to test the accuracy of the classification (the testing examples). In the classical SVM the frontier is defined by a hyper-plane. The frontier is defined so that the examples of the separate categories are divided by a clear gap (the margin in Fig. (1)) that is as wide as possible. New points are automatically predicted to belong to a category based on the side of the gap on which they fall. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

In the next section we are going to explain the classic SVM using a pure optimization perspective. We will ignore the formula of the euclidean distance of a point to a line in $\mathbb{R}^2$ and deduce it from an optimization point of view.

2.1 The Distance of a Point to a Plane as an Optimization Problem

The distance of a point (x_j^1, x_j^2) to a line $r \equiv w_1x_1 + w_2x_2 = b$ is a known algebraic formula that can be obtained by an optimization perspective. We want to find the closest point in r to (x_j^1, x_j^2) according to some measuring distance $D(\cdot)$.

$$\begin{aligned} \min_{x_1,x_2} \; & D\left((x_j^1, x_j^2), (x_1, x_2)\right) \\ s.t. \; & w_1x_1 + w_2x_2 = b \\ & x_1, x_2 \in \mathbb{R} \end{aligned}$$

Using for instance the half of the square of the euclidean distance we obtain:

$$P: \; z = \min \frac{1}{2}\left((x_j^1 - x_1)^2 + (x_j^2 - x_2)^2\right) \tag{1}$$

$$s.t. \; w_1x_1 + w_2x_2 = b \tag{2}$$

$$x_1, x_2 \in \mathbb{R} \tag{3}$$

For a general optimization problem

$$z = \min f(x) \tag{4}$$

$$s.t. \; Ax = b \tag{5}$$

the optimal solution can be found by solving the system

$$\begin{cases} \nabla f = A^T\lambda \\ Ax = \quad b \end{cases} \tag{6}$$

where λ are the Lagrangian multipliers. For problem P we obtain

$$\begin{cases} \nabla f = \begin{bmatrix} (x_j^1 - x_1) \\ (x_j^2 - x_2) \end{bmatrix} = \lambda \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} \\ w_1x_1 + w_2x_2 = b \end{cases} \tag{7}$$

so using (7) in (1) we obtain

$$2z^* = \lambda^2 w_1^2 + \lambda^2 w_2^2 = \lambda^2 \|w\|^2 \tag{8}$$

solving (7) in order of x_1, x_2

$$x_1 = x_j^1 - \lambda w_1 \tag{9}$$

$$x_2 = x_j^2 - \lambda w_2 \tag{10}$$

and replacing it in $w_1x_1 + w_2x_2 = b$

$$w_1(x_j^1 - \lambda w_1) + w_2(x_j^2 - \lambda w_2) = b \tag{11}$$

we obtain

$$\lambda = \frac{w_1x_j^1 + w_2x_j^2 - b}{\|w\|^2} \tag{12}$$

Finally, from (8) and (12) we obtain the distance D_j of a point (x_j^1, x_j^2) to the line $w_1x_1 + w_2x_2 = b$

$$D_j = \sqrt{2z^*} = |\lambda|\|w\| = \frac{|w_1x_j^1 + w_2x_j^2 - b|}{\|w\|} \tag{13}$$

2.2 The SVM Problem

Given a set of training examples (x_j, y_j) for $j \in N = \{1, \ldots, n\}$, corresponding to two classes (one class with $y_j = 1$ for $j \in N_1$ and the other with $y_j = -1$ for $j \in N_2$), in order to find a hyperplane defined by a vector w that maximizes the margin, and such that the points from distinct classes lie on opposite sides of the hyperplane, we need to solve the following optimization problem:

$$SVM: \ v = \max_{j \in N} \min_{w_1, w_2} D_j \tag{14}$$

$$s.t. \ w_1x_j^1 + w_2x_j^2 - b \geq 1 \text{ for } j \in N_1 \tag{15}$$

$$w_1x_j^1 + w_2x_j^2 - b \leq -1 \text{ for } j \in N_2 \tag{16}$$

$$D_j = \frac{|w_1x_j^1 + w_2x_j^2 - b|}{\|w\|} \text{ for } j \in N \tag{17}$$

This maxmin problem can be reformulated as

$$SVM: \ v = \max \delta \tag{18}$$

$$s.t. \ w_1x_j^1 + w_2x_j^2 - b \geq 1 \text{ if } y_j = 1 \tag{19}$$

$$w_1x_j^1 + w_2x_j^2 - b \leq -1 \text{ if } y_j = -1 \tag{20}$$

$$\frac{|w_1x_j^1 + w_2x_j^2 - b|}{\|w\|} \geq \delta \text{ for } j \in N \tag{21}$$

which is equivalent to

$$SVM: \ v = \max \delta \tag{22}$$

$$s.t. \ y_j(w_1x_j^1 + w_2x_j^2 - b) \geq 1 \text{ for } j \in N \tag{23}$$

$$|w_1x_j^1 + w_2x_j^2 - b| \geq \delta\|w\| \text{ for } j \in N \tag{24}$$

Since together with constraints (23) $|w_1 x_j^1 + w_2 x_j^2 - b| = yi(w_1 x_j^1 + w_2 x_j^2 - b)$ we may combine (23) and (24) and obtain

$$SVM: \ v = \max \delta \tag{25}$$

$$s.t. \ y_j(w_1 x_j^1 + w_2 x_j^2 - b) \geq \delta \|w\| \text{ for } j \in N \tag{26}$$

Finally since the problem is invariant to re-scaling of the parameters, we may constraint $\delta\|w\|$ to be equal to any constant in particular 1 and in that case $\delta = \frac{1}{\|w\|}$. Considering that to maximize $\frac{1}{\|w\|}$ is equivalent to minimize $\|w\|$ we obtain the well known SVM problem formulation

$$SVM: \ v = \min \|w\| \tag{27}$$

$$s.t. \ y_j(w_1 x_j^1 + w_2 x_j^2 - b) \geq 1 \text{ for } j \in N \tag{28}$$

There will be some points, in each class for which the margin in Fig. (1) is attained. They are the support vector and are signaled by a double circle in Fig. (1).

3 SVM with Hyperspheres for Classification - A Small Survey

Inspired by SVM [7] proposed by Vapnik, some efforts have been made to improve SVM, in particular, using hyperspheres for classification.

Table 1 resumes some of the papers that constitute a milestone in the development of generalization of SVM based on spheres.

Spherical classifiers were first introduced into pattern classification by Cooper in 1962 [6] and subsequently developed and generalized by many other researchers. Support vector data description (SVDD), [21–23], is an one-class (the target class) classification method. The goal is to devise a closed decision boundary that may distinguish a member of this class from the rest of the feature space, characterized by center and radius $R > 0$ and described by the support vectors. The boundary can also be used to detect novel data or outliers. The model in [21–23] seeks to find a minimum radius hypersphere containing almost all the data from this class. To allow the possibility of outliers in the training set, the distance from data points to the center need not to be strictly smaller than the radius, but these larger distances should be penalized introducing slack variables measuring the distance to the boundary, if an object is outside the description. The sum of the slacks variables is included in the objective function parameterized by a constant as a trade-off between the volume of the hypersphere and the errors.

Different versions of classifiers have been extended from SVDD. These classifiers include maximal-margin spherical-structured multi-class SVM (MSM-SVM) [12], Twin support vector hypersphere (TSVH) [18] and Twin-hypersphere support vector machine (THSVM) [17]. These models find two hyperspheres, one for each class, and classifies points according to which hypersphere a given point

Table 1. Cronological list of papers.

Year	Title	Reference
1999	Support vector domain description	[23]
2002	Uniform object generation for optimizing one-class classifiers	[21]
2004	Support vector data description	[22]
2005	Pattern classification via single-spheres	[24]
2007	A new maximal margin spherical structured multi-class support vector machine	[12]
2009	A fixed-center spherical separation algorithm with kernel transformations	[5]
2009	A new maximal-margin spherical-structured multi-class support	[12]
2010	Least squares twin support vector hypersphere (LS-TSVH) for pattern recognition	[15]
2010	DC models for spherical separation	[3]
2012	Margin maximization in spherical separation	[4]
2013	Sphere Support Vector Machines for large classification tasks	[20]
2013	A twin-hypersphere support vector machine classifier and the fast learning algorithm	[17]
2014	Twin support vector hypersphere (TSVH) classifier for pattern recognition	[18]
2016	A maximum margin and minimum volume hyperspheres machine with pinball loss for imbalanced data classification	[25]
2016	Twin pinball loss support vector hypersphere classifier for pattern recognition	[11]
2017	Steel surface defect classification using multiple hyperspheres support vector machine with additional information	[10]
2017	Linear approach for twin-Hypersphere support vector machine	[13]
2018	Multi-class classification method based on support vector machine with hypersphere for steel surface defects	[9]
2019	Support vector machine with quantile hyperspheres for pattern classification	[14]
2019	A spheres-based support vector machine for pattern classification	[16]

is relatively closest to. The samples in one class are as many as possibly covered by the corresponding hypersphere, whereas the samples in the other class are as far as possible from this hypersphere.

Other approaches related to spherical separation [25] uses maximum margin and minimum volume hyperspheres machine with pinball loss (Pin-M3HM) and least squares twin support vector hyperspheres (LS-TSVH) [15].

A support vector machine with quantile hyperspheres (QHSVM) for binary classification [14] was proposed by Maoxiang et al. (2019). Here, the idea is similar to SVDD, however, it needs to build two hyperspheres with the same center. Considering two boundary hyperspheres (BHSVM) where target samples are inside hyperspheres with radius R+ and negative samples are outside hypersphere with radius R-. BHSVM maximizes the shortest distance between two classes by minimizing the volume of the first and maximizing the volume of the

second, to keep margin as big as possible. QHSVM is generated by introducing the pinball losses in BHSVM.

In this paper, contrary to other approaches we don't minimize the radius. We propose to solve a SVM that is similar to the classic SVM in the sense that we attempt to maximize the margin. The only difference is that the separation boundary is a curve and so it can better adjust to the shape of points as depicted in Fig. 2. Locally, the larger the radius and the further away the center is from the centroid of the points, the closer the boundary is to a line. In this sense it is a generalization of the classic SVM, but that works well even if the data from one class lies in the interior of the second class.

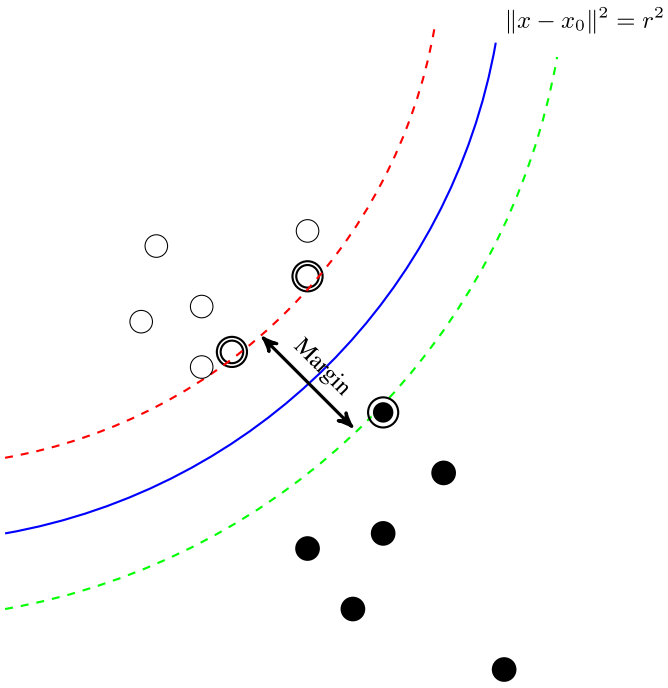


Fig. 2. Non-linear SVM with two features ($\mathbb{R}^2$)

4 A Maximal Margin Hyphersphere SVM Method

Consider a set of $N = \{1, 2, \ldots, n\}$ points each one with $M = \{1, 2, \ldots, m\}$ features and distributed in two separable classes C_1 and C_2, such that $N_1 = \{1, 2, \ldots, |C_1|\}$ and $N_2 = \{1, 2, \ldots, |C_2|\}$.

The goal is to find a maximal separation function $f(x, \theta)$, where θ are the parameters of the function, and $\delta \geq 0$ measures the separation, such that:

$$f(x, \theta) \leq b - \delta \text{ if } x \in C_1$$

$$f(x,\theta) \geq b + \delta \text{ if } x \in C_2.$$

In general we may consider $b = 0$, so to find the optimal θ and δ an optimization problem of the following type must be solved:

$$G: \ v_G = \max h(\delta) \tag{29}$$
$$s.t. f(x_i,\theta) \leq -\delta, \quad \text{if } i \in N_1 \tag{30}$$
$$f(x_j,\theta) \geq +\delta, \quad \text{if } j \in N_2. \tag{31}$$
$$\delta \geq 0 \tag{32}$$

where $h(\delta)$ is a strictly increasing function on $\mathbb{R}^+$.

Spherical SVM have been proposed in the literature to define a non-linear separation between two sets of data with different labels $y = -1$ for $x \in C_1$ and $y = 1$ for $x \in C_2$. The common procedure is to find a sphere centered at some point inside the cloud of points defined by one set, such as the centroid. Sometimes two spheres are proposed each one with center in the centroid of each set. However, these methods will have a poor performance if the points are distributed along a line. In this paper we propose to define a sphere were the center is not necessarily rooted at the center of the points and may even be far outside the centroid if necessary. In this way, depending on the center and the radius, the separation curve may approximate, or not, a linear separator, a margin may be defined as in the linear SVM which we aim to maximize.

4.1 SVM with Spherical Boundary Maximizing the Separation Margin

Our proposal is to find a radius r and a center x_0 of a hypersphere such that the points of one class $\mathcal{C}_1$ fall inside and the points in $\mathcal{C}_2$ fall outside the hypersphere, as in Fig. (2).

$$\|x - x_0\|^2 \leq r^2 \rightarrow x \in C_1$$
$$\|x - x_0\|^2 > r^2 \rightarrow x \in C_2,$$

and there is a margin δ separating the points in each class

$$\|x - x_0\|^2 \leq (r - \delta)^2 \rightarrow x \in C_1$$

$$\|x - x_0\|^2 > (r + \delta)^2 \rightarrow x \in C_2$$

So the goal is to find x_0 and positive r that maximizes non-negative δ.

$$S_0: \ v_0 = \max h(\delta) \tag{33}$$
$$s.t. \|x_i - x_0\|^2 \leq (r - \delta)^2, \quad \text{if } i \in N_1 \tag{34}$$
$$\|x_i - x_0\|^2 \geq (r + \delta)^2, \quad \text{if } i \in N_2 \tag{35}$$
$$\delta, r \geq 0$$

We are going to consider in (33) $h(\delta) = \delta$ and we assign a label $y = \pm 1$ to differentiate the classes. The definition of the label of each class is irrelevant, so we may assume that for $x \in C_1$ we have $y = 1$ and for $x \in C_2$ we have $y = -1$. So from

$$\|x - x_0\|^2 - r^2 - \delta^2 \leq -2r\delta \Rightarrow x \in C_1 \Leftrightarrow y\left(\|x - x_0\|^2 - r^2 - \delta^2\right) \leq -2r\delta \text{ for } y = 1$$

$$\|x - x_0\|^2 - r^2 - \delta^2 > 2r\delta \Rightarrow x \in C_2 \Leftrightarrow y\left(\|x - x_0\|^2 - r^2 - \delta^2\right) \leq -2r\delta \text{ for } y = -1$$

we obtain, the following equivalent problem:

$$S_1 : \ v = \max \delta \tag{36}$$

$$s.t. \ y_i\left(\|x_i - x_0\|^2 - r^2 - \delta^2\right) \leq -2r\delta \text{ for } i \in N \tag{37}$$

Let us consider $z = \begin{bmatrix} x_0 \\ \delta \\ r \end{bmatrix}$, $Q_i = \begin{bmatrix} y_i I(n) & 0 & 0 \\ 0 & -y_i & 1 \\ 0 & 1 & -y_i \end{bmatrix}$, $v_i = \begin{bmatrix} -y_i x_i \\ 0 \\ 0 \end{bmatrix}$.

Let e be a vector of dimension $n + 2$ of zeros except in the $n + 1$ component which is equal to one. S_1 can be rewritten as:

$$S_1 : \ v = \max e^T z \tag{38}$$

$$s.t. \ z^T Q_i z + 2v_i^T z \leq -y_i x_i^T x_i \text{ for } i \in N \tag{39}$$

The Lagrangian function is given by:

$$L(z, \lambda) = e^T z - \sum_{i \in N} \lambda_i \left(z^T Q_i z + 2v_i^T z + y_i x_i^T x_i\right) \tag{40}$$

and

$$\nabla_z L(z, \lambda) = e - \sum_{i \in N} \lambda_i \left(2Q_i z + 2v_i\right) \tag{41}$$

and from $\nabla_z L(z, \lambda) = 0$ we obtain considering $X^y = [y_i x_i]$

$$z = \left(\sum_{i \in N} \lambda_i Q_i\right)^{-1} \left(e - \sum_{i \in N} \lambda_i v_i\right) = \left(\sum_{i \in N} \lambda_i Q_i\right)^{-1} \left(e - V\lambda\right). \tag{42}$$

Considering $\lambda^+ = \sum_{i \in N} \lambda_i$ and $\lambda_y^+ = \sum_{i \in N} y_i \lambda_i$ then

$$\left(\sum_{i \in N} \lambda_i Q_i\right)^{-1} = \begin{bmatrix} \frac{1}{\lambda_y^+} I(n) & 0 & 0 \\ 0 & \frac{\lambda_y^+}{\lambda^{+2} - \lambda_y^{+2}} & \frac{\lambda^+}{\lambda^{+2} - \lambda_y^{+2}} \\ 0 & \frac{\lambda^+}{\lambda^{+2} - \lambda_y^{+2}} & \frac{\lambda_y^+}{\lambda^{+2} - \lambda_y^{+2}} \end{bmatrix}$$

and so

$$z = \begin{bmatrix} x_0 \\ \delta \\ r \end{bmatrix} = \begin{bmatrix} \frac{X^y \lambda}{\lambda_y^+} \\ \frac{\lambda_y^+}{\lambda^{+2} - \lambda_y^{+2}} \\ \frac{\lambda^+}{\lambda^{+2} - \lambda_y^{+2}} \end{bmatrix}$$

and

$$r - \delta = \frac{1}{\lambda^+ + \lambda_y^+} = \frac{1}{\lambda_1^+} \tag{43}$$

$$r + \delta = \frac{1}{\lambda^+ - \lambda_y^+} = \frac{1}{\lambda_2^+} \tag{44}$$

where $\lambda_1^+ = \sum_{i \in N_1} \lambda_i$ and $\lambda_2^+ = \sum_{i \in N_2} \lambda_i$.

Notice that the points x_i for $i \in N$ with $\lambda_i > 0$ are the support vectors. If $i \in N_1$ they fall on the curve $\|x - x_0\|^2 = (r - \delta)^2$ and if $i \in N_2$, they fall in the curve $\|x - x_0\|^2 = (r + \delta)^2$. Since there is at least one support vector by class then the denominators (43) and (44) do not vanish. Also, given that the number of support vectors is small we expect most of the dual variables to be zero. Since the problem is not symmetric, contrary to the linear case, the formulation must be solved twice, considering $y_i = 1$ for $i \in N_1$, $y_i = -1$ for $i \in N_2$ and also for $y_i = -1$ for $i \in N_1$, $y_i = 1$ for $i \in N_2$.

4.2 A Linear Relaxation of the Quadratic Formulation

From the constraints formulation (34) and (35) of S_0 we obtain

$$\|x_i - x_0\|^2 + 2r\delta \leq r^2 + \delta^2, \text{ if } i \in N_1 \tag{45}$$

$$\|x_i - x_0\|^2 - 2r\delta \geq r^2 + \delta^2, \text{ if } i \in N_2 \tag{46}$$

$$\delta, r \geq 0$$

and so from (45) and (46) we have

$$\|x_i - x_0\|^2 + 2r\delta \leq r^2 + \delta^2 \leq \|x_j - x_0\|^2 - 2r\delta \text{ for } i \in N_1, j \in N_2 \Leftrightarrow$$

$$\Leftrightarrow \|x_i\|^2 - 2x_i^T x_0 + \|x_0\|^2 + 2r\delta \leq r^2 + \delta^2 \leq \|x_j\|^2 - 2x_j^T x_0 + \|x_0\|^2 - 2r\delta \text{ for } i \in N_1, j \in N_2$$

$$\Leftrightarrow \|x_i\|^2 - 2x_i^T x_0 + 2r\delta \leq r^2 + \delta^2 - \|x_0\|^2 \leq \|x_j\|^2 - 2x_j^T x_0 - 2r\delta \text{ for } i \in N_1, j \in N_2$$

So, any feasible solution of S_0 verifies the constraints.

$$\|x_i\|^2 - 2x_i^T x_0 + 2r\delta \leq \|x_j\|^2 - 2x_j^T x_0 - 2r\delta \text{ for } i \in N_1, j \in N_2$$

there is

$$4r\delta + 2(x_j - x_i)^T x_0 \leq \|x_j\|^2 - \|x_i\|^2, \text{ for } i \in N_1, j \in N_2$$

Taking $\theta = r\delta$ we obtain the following relaxation problem for S_0.

$$SRL_0 : \max \theta \tag{47}$$

$$\text{s.t. } 4\theta + 2(x_j - x_i)^T x_0 \leq \|x_j\|^2 - \|x_i\|^2, \text{ for } i \in N_1, j \in N_2 \tag{48}$$

$$\theta \geq 0 \tag{49}$$

But this problem may be unbounded, because from

$$4\theta \leq \|x_j\|^2 - \|x_i\|^2 - 2(x_j - x_i)^T x_0,$$

if $2(x_j - x_i)^T x_0 < 0$ then x_0 can be chosen such that $-2(x_j - x_i)^T x_0$ is arbitrarily large. To solve this issue we propose introducing an indirect regularization technique. Instead of adding to the objective function $-k\|x_0\|$ where k is a parameter we introduce a new variable ρ and a new constraint, in the above formulation obtaining:

$$SRL_1(k): \max \theta + k\rho \tag{50}$$

$$\text{s.t. } 4\theta + 2(x_j - x_i)^T x_0 \leq \|x_j\|^2 - \|x_i\|^2 \text{ for } i \in N_1, j \in N_2 \tag{51}$$

$$4\theta + 2(x_j - x_i)^T x_0 \geq \rho \text{ for } i \in N_1, j \in N_2 \tag{52}$$

$$\theta \geq 0 \tag{53}$$

The next example illustrates the application of the quadratic and linear formulation.

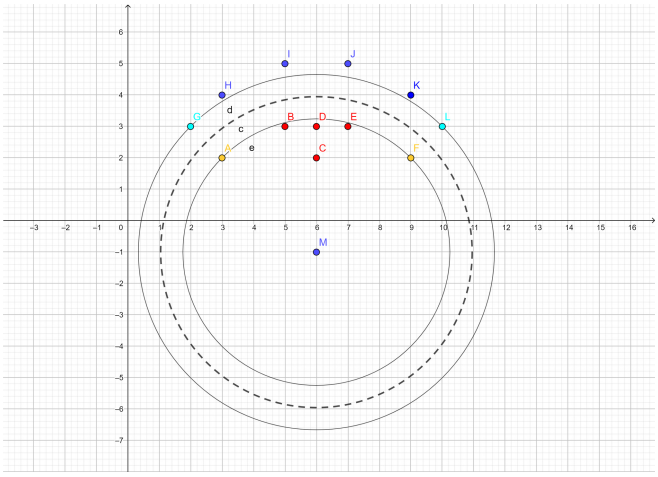


Fig. 3. Spherical separation using the quadratic formulation

Example 1. Consider the set of points in $\mathbb{R}^2$ from two classes, each with 6 elements.

Class1

x_1	3	5	6	6	7	9
x_2	2	3	2	3	3	2

Class2

x_1	2	3	5	7	9	10
x_2	3	4	5	5	4	3

For the quadratic formulation we obtained $x_0 = (6, -1)$, $r = 4.9497$ and the margin, $\delta = 0.7071$. The support vectors are $(3, 2)$ and $(9, 2)$ for Class 1 and $(2, 3)$ and $(10, 3)$ for Class 2. This solution is depicted in Fig. 3.

For the linear formulation we obtained $x_0 = (6, -1.5)$, $\theta = 3.75$, $r = 5.3153$ and the margin, $\delta = 0.7055$. The support vectors are $(1, 2)$, $(5, 3)$, $(7, 3)$ and $(9, 2)$ for Class 1 and $(2, 3)$ and $(10, 3)$ for Class 2 . This solution is depicted in Fig. 4.

5 Computational Experience

The goal of the numerical experience is to test the accuracy of the quadratic (38–39 in Sect. 4.1) and linear (50–53 in Sect. 4.2) models and compare against the classical SVM approach for real and generated data.

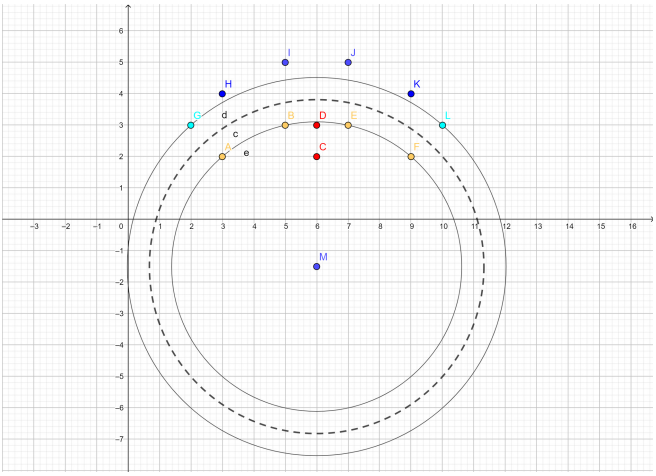


Fig. 4. Spherical separation using the linear formulation

Experiments were run on a computer with an Intel(R) Core(TM) i7-8550U processor, 1.80 GHz, 8.00 GB of RAM, 64-bit operating system, running on Windows 10, version 20H2. We code our method on Matlab R2019a, and we used BARON software [19] version 13.0.1, running on Matlab with MATLAB/BARON interface version v1.69. The BARON solver was used to solve the quadratic formulation. To run the classical SVM we used the Statistics and Machine Learning Toolbox from Matlab version 11.5.

The real data used was the *Seeds* data set, available from UCI Machine Learning Repository [1]. This data set considers three different varieties of wheat with 70 elements each, and the set is constructed with measures of seven geometric attributes. In our experiments we used only two spherically separated sets (140

objects). We performed 20 trials, choosing in each one, randomly, a set for training with 80 objects (40 of each class) and a set for testing with 60 objects (30 of each class).

We build also an artificial data set in order to have two sets, spherically separated, each one with three attributes. So, for set A we consider 50 points $a_i = (a_{i_1}, a_{i_2}, a_{i_3})$ so that: a_{i_1}satisfies a Gaussian distribution $N(2, 2)$, a_{i_2}satisfies a Gaussian distribution $N(5, 2)$ and $a_{i_3} = a_{i_1}^2 + a_{i_2}^2$. For set B we considered also 50 points $b_j = (b_{j_1}, b_{j_2}, b_{j_3})$ so that: b_{j_1}satisfies a Gaussian distribution $N(2, 1)$, b_{j_2}satisfies a Gaussian distribution $N(3, 1)$ and $b_{j_3} = 40 + b_{i_1}^2 + b_{i_1 2}^2$. From the 50 elements we randomly chose 25 points as the training set, the remaining 25 being the test set. As for the *Seeds* database we performed 20 trials.

Table 2. Accuracy results

	SVM model			
Data set	P_1	Quadratic	Linear	Classic (MATLAB)
	Artificial	99,20	95,30	95,70
	Seeds	97,17	96,25	95,42

Given a specific classification method, for every trial i, $i = 1, \ldots, 20$, considering that the test set has $n_i^1 = n_{1i}^1 + n_{0i}^1$ elements in class 1 and $n_i^2 = n_{1i}^2 + n_{0i}^2$ elements in class 2, the well classified element in class 1 and 2 are given by n_{1i}^1 and n_{1i}^2 respectively, and the number of misclassified objects is given by n_{0i}^1 and n_{0i}^2, for class 1 and 2 respectively. The accuracy of each method can be inferred by studying the proportion of well classified objects P_1 or of misclassified objects P_0, given by,

$$P_1 = \frac{1}{20}\sum_{i=1}^{20}\frac{n_{1i}^1 + n_{1i}^2}{n^1 + n^2}, \quad P_0 = \frac{1}{20}\sum_{i=1}^{20}\frac{n_{0i}^1 + n_{0i}^2}{n^1 + n^2}.$$

Table 2 presents the overall proportion of correctly classified objects P_1, for the testing sets in the 20 trials. Since P_0 can be obtained directly from P_1 there is no need to explicitly present it.

The results for both data sets show a better classification performance when using the quadratic model. For the linear model, the performance was slightly worse with the artificial data and better with the *Seeds* data set, when compared to the classic SVM. To test the statistical significance of the difference between the accuracy of two models, we used the Resampled Paired t Test [8]. The test statistic is given by,

$$t = \frac{\sqrt{20}\hat{p}}{\sqrt{\frac{\sum_{i=1}^{2}(p(i)-\hat{p})^2}{n-1}}} \tag{54}$$

where $\bar{p} = \frac{\sum_{i=1}^{n}(p^{(i)})}{n}$ and $p^{(i)}$ is the difference, in trial i, between the proportion of well classified objects of the two methods. The null hypothesis states the

equality of the accuracy of both methods against the alternative hypothesis of that one of them is more accurate than the other. Table 3 and 4 present the result of the statistical tests for the real and generated sets respectively. Globally there is only statistical evidence of the out performance of the Quadratic model in comparison with the Classic model.

Table 3. Statistical hypothesis test for the *Seeds* data

Alternative hypothesis	$\bar{p}$	Statistic t	p-value
Quadratic more accurate than Classic	0,018	3,367	0,002
Quadratic more accurate than Linear	0,009	1,764	0,047
Linear more accurate than Classic	0,008	1,365	0,094

Table 4. Statistical hypothesis test for the generated data

Alternative hypothesis	$\bar{p}$	Statistic t	p-value
Quadratic more accurate than Classic	0,035	5,872	0,000
Quadratic more accurate than Linear	0,039	5,224	0,000
Classic more accurate than Linear	0,004	0,395	0,348

6 Conclusions

In this paper we presented a formulation with a hypersphere separation, maximizing the margin. The formulation is a generalization of the classical SVM. We derived a linear relaxation for this problem and developed a regularization procedure for avoiding unboundedness. The preliminary results seem promising, and we obtained a good approximation for the quadratic solution computationally cheaper since it is based on a linear formulation. We intend to explore in the future the dual problem and to adjust the formulation to non-separable classes and multi-class classification.

Acknowledgments. We would like to thank the reviewers for their valuable comments and suggestions that helped to improve our manuscript.

References

1. Uc irvine machine learning repository. https://archive-beta.ics.uci.edu/
2. Abe, S.: Support Vector Machines for Pattern Classification. Springer, London (2005). https://doi.org/10.1007/1-84628-219-5
3. Astorino, A., Fuduli, A., Gaudioso, M.: A fixed-center spherical separation algorithm with kernel transformations for classification problems. J. Glob. Optim. **48**, 657–669 (2010). https://doi.org/10.1007/s10898-010-9558-0

4. Astorino, A., Fuduli, A., Gaudioso, M.: Margin maximization in spherical separation. Comput. Optim. Appl. **53**, 301–322 (2012). https://doi.org/10.1007/s10589-012-9486-7
5. Astorino, A., Gaudioso, M.: A fixed-center spherical separation algorithm with kernel transformations for classification problems. Comput. Manag. Sci. **6**, 357–372 (2009). https://doi.org/10.1007/s10287-007-0051-2
6. Cooper, P.W.: The hypersphere in pattern recognition. Inf. Control **5**(4), 324–346 (1962). https://doi.org/10.1016/S0019-9958(62)90641-1, https://www.sciencedirect.com/science/article/pii/S0019995862906411
7. Corinna, C., Vladimir, V.: Support-vector networks. Mach. Learn. **20**, 273–297 (1995). https://doi.org/10.1007/BF00994018
8. Dietterich, T.G.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Comput. **10**(7), 1895–1923 (1998). https://doi.org/10.1162/089976698300017197
9. Gong, R., Wu, C., Chu, M.: Multi-class classification method based on support vector machine with hyper-sphere for steel surface defects. In: 2018 37th Chinese Control Conference (CCC), pp. 9197–9202 (2018). https://doi.org/10.23919/ChiCC.2018.8483656
10. Gong, R., Wu, C., Chu, M.: Steel surface defect classification using multiple hyperspheres support vector machine with additional information. Chemom. Intell. Lab. Syst. **172**, 109–117 (2017). https://doi.org/10.1016/j.chemolab.2017.11.018
11. Gong, R., Wu, C., Chu, M., Wang, H.: Twin pinball loss support vector hypersphere classifier for pattern recognition. In: 2016 Chinese Control and Decision Conference (CCDC), pp. 6551–6556 (2016). https://doi.org/10.1109/CCDC.2016.7532177
12. Hao, P.Y., Chiang, J.H., Lin, Y.H.: A new maximal-margin spherical-structured multi-class support vector machine. Appl. Intell. **30**(2), 98–111 (2009). https://doi.org/10.1007/s10489-007-0101-z
13. Ketabchi, S., Moosaei, H., Razzaghi, M.: Linear approach for twin-hypersphere support vector machine. Adv. Model. Optim. **19**, 79–85 (2017)
14. Chu, M., Liu, X., Gong, R., Zhao, J.: Support vector machine with quantile hyperspheres for pattern classification. PLoS ONE **14**(2) (2019). https://doi.org/10.1371/journal.pone.0212361
15. Peng, X.: Least squares twin support vector hypersphere (LS-TSVH) for pattern recognition. Expert Syst. Appl. **37**(12), 8371–8378 (2010). https://doi.org/10.1016/j.eswa.2010.05.045, https://www.sciencedirect.com/science/article/pii/S0957417410004562
16. Peng, X.: A spheres-based support vector machine for pattern classification. Neural Comput. Appl. **31**(1), 379–396 (2017). https://doi.org/10.1007/s00521-017-3004-x
17. Peng, X., Xu, D.: A twin-hypersphere support vector machine classifier and the fast learning algorithm. Inf. Sci. **221**, 12–27 (2013). https://doi.org/10.1016/j.ins.2012.09.009, https://www.sciencedirect.com/science/article/pii/S0020025512005919
18. Peng, X., Xu, D.: Twin support vector hypersphere (TSVH) classifier for pattern recognition. Neural Comput. Appl. **24**(5), 1207–1220 (2013). https://doi.org/10.1007/s00521-012-1306-6
19. Sahinidis, N.V.: BARON 21.1.13: Global Optimization of Mixed-Integer Nonlinear Programs, User's Manual (2017)
20. Strack, R., Kecman, V., Strack, B., Li, Q.: Sphere support vector machines for large classification tasks. Neurocomputing **101**, 59–67 (2013). https://doi.org/10.1016/j.neucom.2012.07.025

21. Tax, D.M.J., Duin, R.P.W.: Uniform object generation for optimizing one-class classifiers. J. Mach. Learn. Res. **2**, 155–173 (2002)
22. Tax, D.M.J., Duin, R.P.W.: Support vector data description. Mach. Learn. **54**(1), 45–66 (2004)
23. Tax, D.M., Duin, R.P.: Support vector domain description. Pattern Recognit. Lett. **20**(11), 1191–1199 (1999). https://doi.org/10.1016/S0167-8655(99)00087-2, https://www.sciencedirect.com/science/article/pii/S0167865599000872
24. Wang, J., Neskovic, P., Cooper, L.N.: Pattern classification via single spheres. In: Hoffmann, A., Motoda, H., Scheffer, T. (eds.) DS 2005. LNCS (LNAI), vol. 3735, pp. 241–252. Springer, Heidelberg (2005). https://doi.org/10.1007/11563983_21
25. Xu, Y., Yang, Z., Zhang, Y., Pan, X., Wang, L.: A maximum margin and minimum volume hyper-spheres machine with pinball loss for imbalanced data classification. Knowl.-Based Syst. **95**, 75–85 (2016). https://doi.org/10.1016/j.knosys.2015.12.005, https://www.sciencedirect.com/science/article/pii/S0950705115004773