



Spherical separation with infinitely far center

Annabella Astorino¹ · Antonio Fuduli²

Published online: 7 October 2020
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

Abstract

We tackle the problem of separating two finite sets of samples by means of a spherical surface, focusing on the case where the center of the sphere is fixed. Such approach reduces to the minimization of a convex and nonsmooth function of just one variable (the radius), revealing very effective in terms of computational time. In particular, we analyze the case where the center of the sphere is selected far from both the two sets, embedding the grossone idea and obtaining a kind of linear separation. Some numerical results are presented on classical binary data sets drawn from the literature.

Keywords Spherical separation · Classification · Grossone

1 Introduction

Classification problems in mathematical programming concern separation of sample sets by means of an appropriate surface. This field, entered by many researchers in optimization community in the last years, is a part of the more general machine learning area, aimed at providing automated systems able to learn from human experiences.

In machine learning, classification can be addressed on the basis of different paradigms (Astorino et al. 2008). The most common one is the supervised approach, where the samples in each set are equipped with the class label heavily exploited in the learning phase. A well-established supervised technique is the support vector machine (SVM) (Cristianini and Shawe-Taylor 2000; Vapnik 1995), which revealed a powerful classification tool in many applicative areas. A widely adopted alternative is called unsupervised, since no class label is known in advance; the aim is to cluster the data on the basis of their similarities (Celebi 2015). In the middle, we find the semisupervised techniques (Chapelle et al. 2006), which are a compromise between the supervised and

the unsupervised approaches; in such case, the learning task is characterized by the exploitation of the overall information coming from both labeled and unlabeled samples. Some useful references are Chapelle and Zien (2005) and Astorino and Fuduli (2007), the latter being a semisupervised version of the SVM technique.

A more recent classification framework is constituted by the multiple instance learning (MIL) (Herrera et al. 2016), which can be interpreted as a kind of weak supervised approach; it consists in categorizing bags of samples, being available only the class label of the bags instead of the class label of each sample inside them. A seminal SVM-type MIL paper is Andrews et al. (2003), while some recent articles are Astorino et al. (2018, 2019a, b, 2020), Avolio and Fuduli (2020), Gaudioso et al. (2020), and Plastria et al. (2014).

In this work, we present an extension of the supervised binary classification approach reported in Astorino and Gaudioso (2009) and based on the spherical separation of two finite sets of samples (points in \mathbb{R}^n), say

$$\mathcal{A} = \{a_1, \dots, a_m\}, \text{ with } a_i \in \mathbb{R}^n, i = 1, \dots, m$$

and

$$\mathcal{B} = \{b_1, \dots, b_k\}, \text{ with } b_l \in \mathbb{R}^n, l = 1, \dots, k.$$

As initially proposed in Tax and Duin (1999) and also in Astorino and Gaudioso (2009), the objective is to find a minimal volume sphere separating the two sets \mathcal{A} and \mathcal{B} , i.e., a sphere enclosing all points of \mathcal{A} and no points of \mathcal{B} . In particular, while in Tax and Duin (1999) the optimization

Communicated by Yaroslav D. Sergeyev.

✉ Annabella Astorino
annabella.astorino@icar.cnr.it

Antonio Fuduli
antonio.fuduli@unical.it

¹ ICAR - National Research Council, Rende, Italy

² Department of Mathematics and Computer Science, University of Calabria, Rende, Italy

problem to be solved is characterized by $n + 1$ unknowns, the center of the sphere in \mathbb{R}^n and the radius in \mathbb{R} , in Astorino and Gaudioso (2009) the center is prefixed; consequently, in such case, the problem becomes univariate since it reduces to just the computation of one variable, the radius. For this problem, the authors have designed a computationally fast algorithm which, despite the drastic introduced simplification, provides reasonably good separation results, as long as the center is judiciously chosen. Possible choices for the center are, for example, the barycenter of one of the two sets, \mathcal{A} or \mathcal{B} , or the barycenter of the set $\mathcal{A} \cup \mathcal{B}$ when there is no information on the geometry of the data.

In this paper, we use the same approach as in Astorino and Gaudioso (2009), computing the optimal radius of the separation sphere when the center is prefixed. The novelty of our proposal consists in selecting the center infinitely far from the two sets, exploiting the new numeral system based on the grossone theory (Sergeyev 2017).

Spherical separation falls into the class of the nonlinear separation surfaces (Astorino et al. 2008, 2016), differently, for example, from the well-known SVM technique (Cristianini and Shawe-Taylor 2000; Vapnik 1995), where a classifier is constructed by generating a hyperplane far away from the points of the two sets. Also the SVM approach allows to obtain general nonlinear classifiers by adopting kernel transformations. In this case, the basic idea is to map the data into a higher-dimensional space (the feature space) and to separate the two transformed sets by means of one hyperplane, that corresponds to a nonlinear surface in the original input space. The main advantage of spherical separation is that once the center of the sphere is heuristically fixed in advance, the optimal radius can be found quite effectively by means of a simple sorting algorithm such as in Astorino and Gaudioso (2009) and Astorino et al. (2012b). No analogous simplification strategy is apparently available if one adopts the SVM approach. Moreover, another advantage is to work directly in the input space. In fact, to keep, whenever possible, the data in the original space seems appealing in order to stay close to the real-life modeled processes. Of course, kernel methods are characterized by high flexibility, even if sometimes they provide results which are hard to be interpreted in the original input space, differently from the nonlinear classifiers acting directly in such space (see, e.g., Astorino et al. (2014b, c)).

The paper is organized in the following way. In the next section, we recall the main concepts related to linear and spherical separabilities; in fact, the linear separation surface (i.e., a hyperplane) can be interpreted as a sphere characterized by an infinitely far center and, consequently, by an infinite length radius. In Sect. 3, we summarize the main concepts of the grossone algebra, and in Sect. 4, we describe our approach, equipped with some numerical results obtained

on a set of benchmark test problems drawn for the literature. Finally, some conclusions are reported in the last section.

Throughout the paper, we use the following notation. We indicate by $\|\cdot\|$ the Euclidean norm, and given a set \mathcal{X} , we denote by $\text{conv}(\mathcal{X})$ the convex hull of \mathcal{X} .

2 Linear and spherical separation

A seminal paper on linear separation appeared in 1965 by Mangasarian (1965), while the first approach for pattern classification based on a minimum volume sphere dates back to 1999 by Tax and Duin (1999).

2.1 Linear separability

Two sets \mathcal{A} and \mathcal{B} are linearly separable if and only if there exists a hyperplane

$$H(w, \gamma) = \{x \in \mathbb{R}^n \mid w^T x = \gamma\}, \text{ with } w \in \mathbb{R}^n \text{ and } \gamma \in \mathbb{R},$$

such that

$$w^T a_i \leq \gamma - 1 \quad i = 1, \dots, m$$

and

$$w^T b_l \geq \gamma + 1 \quad l = 1, \dots, k.$$

A characterization of linear separability is given by the following condition:

$$\text{conv}(\mathcal{A}) \cap \text{conv}(\mathcal{B}) = \emptyset,$$

which is well depicted in Fig. 1, where the two cases of linearly separable and inseparable sets are considered, respectively.

In linear separability, a relevant role is played by the SVM technique (Cristianini and Shawe-Taylor 2000; Vapnik 1995), which provides a classifier characterized by a good generalization capability, i.e., the ability to correctly classify a new sample point. This approach consists in constructing a separation hyperplane far away from the points of both the two sets \mathcal{A} and \mathcal{B} , by minimizing the following error function:

$$\begin{aligned} \min_{w, \gamma} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m \max\{0, a_i^T w - \gamma + 1\} \\ + C \sum_{l=1}^k \max\{0, -b_l^T w + \gamma + 1\}, \end{aligned}$$

where the minimization of first term corresponds to the maximization of the margin (i.e., the distance between two parallel

Fig. 1 Linear separability: (a) separable sets; (b) inseparable sets

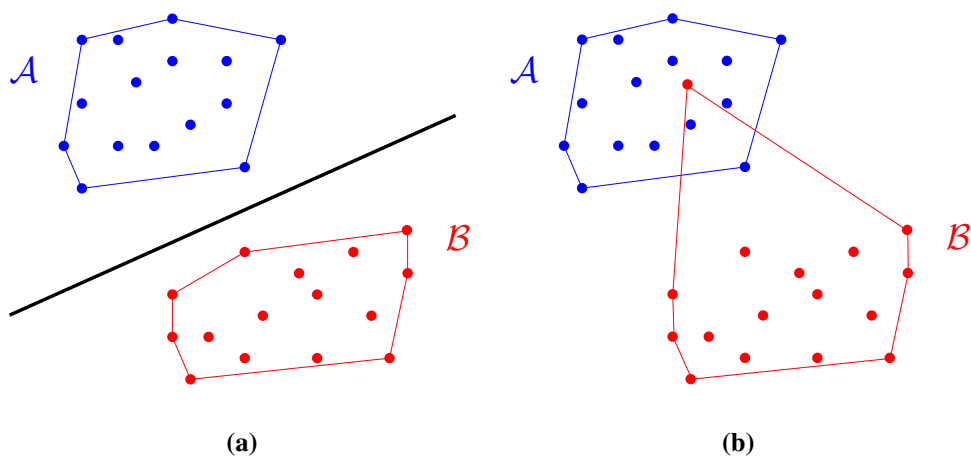
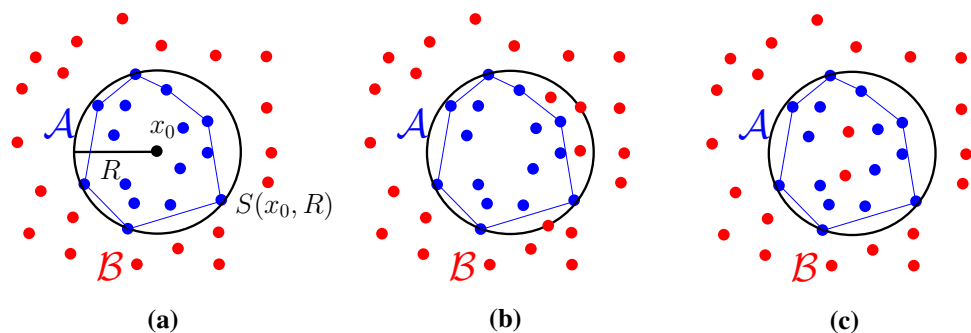


Fig. 2 Spherical separability: (a) separable sets; (b), (c) inseparable sets



hyperplanes supporting the sets), while the last two terms represent the misclassification errors in correspondence to the two point sets \mathcal{A} and \mathcal{B} , respectively. The parameter C is a positive constant giving the trade-off between these two objectives.

2.2 Spherical separability

In the spherical separation case, the pursued idea is to find a sphere

$$S(x_0, R) = \{x \in \mathbb{R}^n \mid \|x - x_0\|^2 = R^2\},$$

with center $x_0 \in \mathbb{R}^n$ and radius R , enclosing all points of \mathcal{A} and no points of \mathcal{B} . More formally, the two sets \mathcal{A} and \mathcal{B} are spherically separated by $S(x_0, R)$ if and only if

$$\|a_i - x_0\|^2 \leq R^2 \quad i = 1, \dots, m$$

and

$$\|b_l - x_0\|^2 \geq R^2 \quad l = 1, \dots, k.$$

We observe that in this case, the role played by the two sets is not symmetric; in fact, a necessary (but not sufficient) condition for the existence of a separation sphere is the following (Fig. 2):

$$\text{conv}(\mathcal{A}) \cap \mathcal{B} = \emptyset.$$

Based on the above spherical separability definition, the classification error associated with any sphere $S(x_0, R)$ is

$$\sum_{i=1}^m \max\{0, \|a_i - x_0\|^2 - R^2\} + \sum_{l=1}^k \max\{0, R^2 - \|b_l - x_0\|^2\}.$$

To take into account the generalization capability, in Astorino and Gaudioso (2009) the authors proposed to construct a minimal volume separation sphere by solving the following problem:

$$\begin{aligned} \min_{x_0, z} z + C \sum_{i=1}^m \max\{0, \|a_i - x_0\|^2 - z\} \\ + C \sum_{l=1}^k \max\{0, z - \|b_l - x_0\|^2\}, \end{aligned} \tag{1}$$

with $z \triangleq R^2 \geq 0$ and $C > 0$ being the parameter tuning the trade-off between the minimization of the volume and the minimization of the classification error.

Some works devoted to spherical separation are Astorino and Gaudioso (2009), Astorino et al. (2010, 2012a, b, 2014a, 2017), and Le Thi et al. (2013). In particular, the approach we propose in this paper is based on the fixed-center algorithm introduced in Astorino and Gaudioso (2009), where

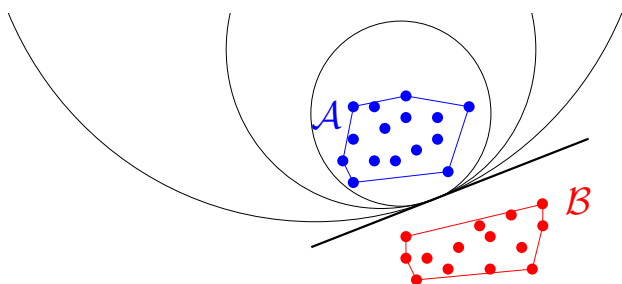


Fig. 3 Linear separability versus spherical separability

the center x_0 of the sphere is assumed to be fixed (e.g., equal to the barycenter of \mathcal{A}). Indicating by p the cardinality of the biggest set between \mathcal{A} and \mathcal{B} , it is easy to see that whenever x_0 is fixed, problem (1) reduces to a univariate, convex, nonsmooth optimization problem, and it is rewritable as a structured linear program, whose dual can be solved in time $O(p \log p)$. In fact, the optimal value of the variable z (the square of the radius) is computable by simply comparing the distances, preliminarily sorted, between the center x_0 and each point in the two sets. For further technical details on such approach, we refer the reader directly to Astorino and Gaudioso (2009).

2.3 Spherical separation versus linear separation

From the mathematical point of view, both the approaches (linear and spherical separations) are characterized by the same number of parameters to be computed; in fact, a separation hyperplane is identified by the bias and the normal, while a sphere is computed by determining the center and the radius. In this perspective, a hyperplane can be viewed as a particular sphere where the center is infinitely far (Fig. 3).

A possible choice of the center x_0 is to take a point far from both the sets \mathcal{A} and \mathcal{B} , i.e.,

$$x_0 = x_0^A + M(x_0^A - x_0^B), \tag{2}$$

where

$$x_0^A \triangleq \frac{1}{m} \sum_{i=1}^m a_i \quad \text{and} \quad x_0^B \triangleq \frac{1}{k} \sum_{l=1}^k b_l$$

are the barycenters of \mathcal{A} and \mathcal{B} , respectively, while M is a sufficiently large positive parameter, commonly named “big M ”.

Formula (2) corresponds to computing x_0 from x_0^A along the direction $x_0^A - x_0^B$ with stepsize equal to M (Fig. 4).

Notice that in general, the “big M ” constant is not easy to be managed from the numerical point of view, since indeed it is not evident how to quantify the minimum threshold value

such that M could be considered sufficiently big; as a consequence, in the practical cases, the necessity to test many trial values arises. A possible way to overcome this numerical difficulty is to obtain an infinitely far center by exploiting the new grossone theory and just setting M equal to $\textcircled{1}$. The symbol $\textcircled{1}$ denotes the new numeral, called *grossone*, which is the object of the next section. Differently from Astorino and Gaudioso (2009), where various values of M in formula (2) have been tested in order to obtain a good classification performance, a remarkable advantage in using the grossone resides in avoiding the necessity to repeat several tests with larger and larger values of M .

3 Generality on the grossone algebra

Grossone, denoted by the symbol $\textcircled{1}$, has been introduced as a basic element of a new numeral system thanks to which it is possible to express not only finite but also infinite and infinitesimal quantities and to execute numerical computations with all of them, in a unique framework with finite quantities. An explicative recent survey is Sergeyev (2017).

We remark that this new computational methodology is not related to the nonstandard analysis (Sergeyev 2019), and it is noncontradictory as studied in depth in Lolli (2015), Margenstern (2011), Montagna et al. (2015). Moreover, a new supercomputer based on grossone was conceived; it is called *Infinity Computer* and is patented in several countries (Sergeyev 2010a).

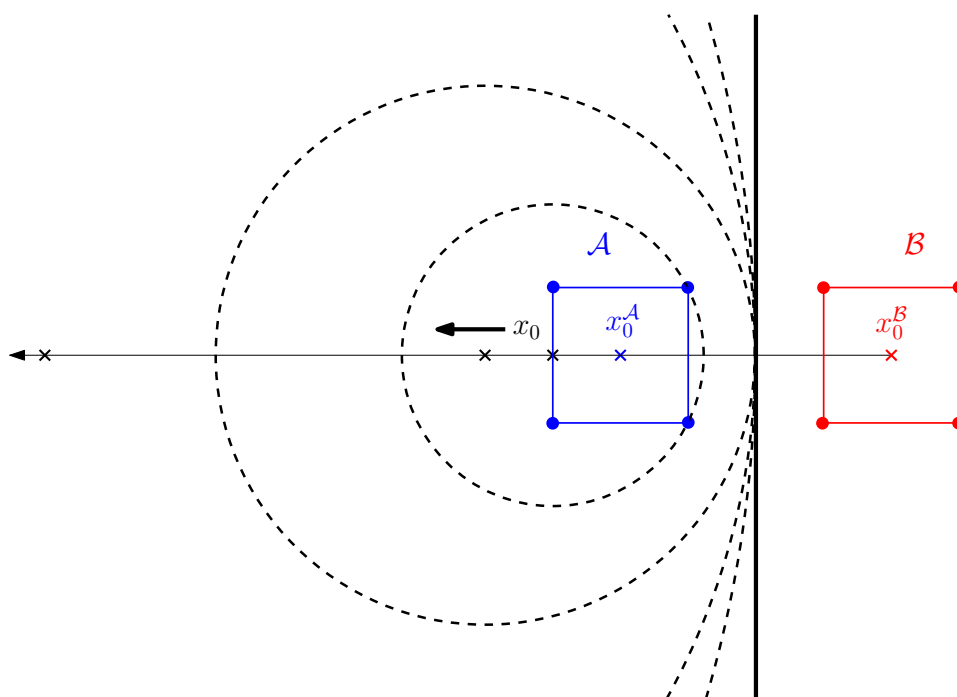
In the literature, there are a lot of applications of grossone in various fields: in optimization (Cococcioni et al. 2020; De Cosmis and De Leone 2012; De Leone 2018; De Leone et al. 2020, 2018; Gaudioso et al. 2018; Sergeyev et al. 2018), in numerical differentiation (Sergeyev 2011), in ordinary differential equations (Iavernaro et al. 2020) and in many other theoretical and computational research areas such as infinite series (Caldarola 2018; Sergeyev 2009, 2017, 2018; Zhigljavsky 2012). On the other hand, to the best of our knowledge, it seems that there is still no paper involving the grossone theory in classification problems.

Grossone is an infinite unit, defined as the number of the elements in the set \mathbb{N} of the natural numbers. The new numeral $\textcircled{1}$ is introduced by describing its properties postulated in the *infinite unit axiom*:

1. *Infinity*: Any finite natural number n is less than $\textcircled{1}$.
2. *Identity*: $0 \cdot \textcircled{1} = \textcircled{1} \cdot 0 = 0$, $\textcircled{1} - \textcircled{1} = 0$, $\frac{\textcircled{1}}{\textcircled{1}} = 1$, $\textcircled{1}^0 = 1$, $1^{\textcircled{1}} = 1$, $0^{\textcircled{1}} = 0$.
3. *Divisibility*: For any finite natural number n , the sets

$$\mathbb{N}_{k,n} = \{k, k + n, k + 2n, k + 3n, \dots\}, \quad 1 \leq k \leq n, \quad \bigcup_{k=1}^n \mathbb{N}_{k,n} = \mathbb{N}$$

Fig. 4 Spherical separability with a far center



have a number of elements indicated by $\frac{\textcircled{1}}{n}$.

In particular, this axiom states that for any given finite integer n , the infinite number $\frac{\textcircled{1}}{n}$ is integer being larger than any finite number. In addition to the standard axioms of the real numbers, it allows to maintain all the basic properties such as commutative or associative.

A general way to express infinite and infinitesimal numbers on a computer is provided in Sergeyev (2003, 2010b, 2015, 2017) by using a numeral positional system with the infinite base $\textcircled{1}$. A number Q in this new numeral system can be represented by groups of powers of $\textcircled{1}$:

$$Q = q_{p_h} \textcircled{1}^{p_h} + \dots + q_{p_1} \textcircled{1}^{p_1} + q_{p_0} \textcircled{1}^{p_0} + q_{p_{-1}} \textcircled{1}^{p_{-1}} + \dots + q_{p_{-r}} \textcircled{1}^{p_{-r}}, \tag{3}$$

where

- h and r are integer numbers ($\in \mathbb{N}$);
- the exponents p_i ($i = -r, \dots, -1, 0, 1, \dots, h$), called gross-powers, can be in turn numbers of the same type as Q and they are sorted in the decreasing order

$$p_h > p_{h-1} > \dots > p_1 > p_0 > p_{-1} > \dots > p_{-(r-1)} > p_{-r},$$

with $p_0 = 0$;

- $q_{p_i} \neq 0$ ($i = -r, \dots, -1, 0, 1, \dots, h$), called gross-digits, are finite, positive or negative numbers.

Some explicative examples of number representations in this numeral system are the following.

- Finite numbers are represented by numerals with the highest gross-power equal to zero, e.g., $-7.3 = -7.3 \textcircled{1}^0$.
- Infinitesimal numbers are represented by numerals having negative finite or infinite gross-powers. The simplest infinitesimal is $\textcircled{1}^{-1} = \frac{1}{\textcircled{1}}$ for which $\textcircled{1}^{-1} \cdot \textcircled{1} = 1$. Note that all infinitesimals are not equal to zero. In particular, $\textcircled{1}^{-1} > 0$ because it is a result of division of two positive numbers.
- Infinite numbers have at least one positive finite or infinite gross-power. For instance, the number $23.65 \textcircled{1}^{41.72} + 45.13 \textcircled{1}^{30.6} - 12.27 \textcircled{1}^{-22.1}$ is infinite; it consists of two infinite parts and one infinitesimal part.

We conclude remarking that, by using the grossone numeral system, it is easy to manage all types of computations, since it is allowed to assign infinite and infinitesimal values to quantities.

4 Computational experiments

We have tested the fixed-center spherical separation algorithm described in Astorino and Gaudioso (2009) for solving problem (1), by choosing the center of the sphere as follows:

$$x_0 = x_0^A + \textcircled{1} (x_0^A - x_0^B),$$

Table 1 Data sets

Data set	Dimension	Points
Cancer	9	699
Diagnostic	30	569
Heart	13	297
Pima	9	769
Ionosphere	34	351
Sonar	60	208
Mushrooms	22	8124
Prognosis	32	110
Tic Tac Toe	9	958
Votes	16	435
Galaxy	14	4192
g50c	50	550
g10n	10	550

i.e., by setting $M = \textcircled{1}$ in formula (2).

The code, named $FC_{\textcircled{1}}$ (fixed center—infinately far), has been implemented in MATLAB (version R2017b), and it has been run on a Windows 10 system, characterized by a 2.21 GHz processor and 16 GB of RAM. It has been tested on 13 data sets drawn from the literature and listed in Table 1. The first ten test problems are taken from the UCI machine learning repository (Murphy and Aha 1992), a collection of databases, domain theories, and data generators that are used by the machine learning community. Galaxy is the data set used in galaxy discrimination with neural networks (Ode-wahn et al. 1992), while a detailed description of g50c and g10n is reported in Chapelle and Zien (2005).

For managing the grossone arithmetic operations, we have used the MATLAB environment of the new Simulink-based solution of the Infinity Computer (Falcone et al. 2020), where an arithmetic C++ library is integrated within a MATLAB environment. In particular, given the two gross-numbers x and y , from such library we have used the following C++ subroutines:

- $\text{TestGrossMatrix}(x, y, '-')$, returning the difference between x and y ;
- $\text{TestGrossMatrix}(x, y, '+')$, returning the sum of x and y ;
- $\text{TestGrossMatrix}(x, y, '*')$, returning the product of x and y ;
- $\text{GROSS_cmp}(x, y)$, returning 1 if $x > y$, -1 if $x < y$ and 0 if $x = y$.

Using the MATLAB notation, we have expressed any vector g of n gross-number elements (that in the sequel, for the sake

of simplicity, we call gross-vector) as a couple (G, fg) , with

$$G = [g_1; g_2; \dots; g_n] \quad \text{and} \quad fg = [fg_1 \ fg_2 \dots fg_n],$$

where $g_j, j = 1, \dots, n$, is an array of dimension $s \times 2$ representing a gross-number Q , with $s = h + r + 1$ (see formula (3)). For each row of g_j , the first element contains a gross-digit, while the second one contains the corresponding gross-power. Since s depends on r and h , which can be different for each array g_j of the same gross-vector g , the scalar $fg_j, j = 1, \dots, n$, is necessary to provide the position in G of the last component of g_j .

To manage the gross-vectors, we have also implemented the following new MATLAB subroutines:

- $\text{realToGrossone}(r)$, returning a grossone representation (G, fg) of a real vector r ;
- $\text{extract}(G, fg, i)$, returning the i th gross-number in the gross-vector (G, fg) ;
- $\text{normGrossone}(G, fg)$, computing the squared Euclidean norm of the gross-vector (G, fg) ;
- $\text{scalProdG}(G_1, fg_1, G_2, fg_2)$, computing the scalar product between the two gross-vectors (G_1, fg_1) and (G_2, fg_2) ;
- $\text{BubbleSortGrossone}(G, fg, \text{sign})$, sorting the gross-vector (G, fg) in the ascending order if $\text{sign} = 1$ and in the descending order if $\text{sign} = -1$.

For each data set, in order to compute the best value of the parameter C , we have adopted a bilevel cross-validation strategy (Astorino and Fuduli 2016), by varying of C in the grid $\{10^{-1}, 10^0, 10^1, 10^2\}$; such choice of the grid has been suggested by the necessity to obtain a nonzero optimal value of z , which in turn provides the optimal value of the radius R , as shown in Astorino and Gaudioso (2009) where the authors have proved the following proposition.

Proposition 1 *The following statements hold:*

- (i) if $C < 1/m$, then $z^* = 0$;
- (ii) if $C > 1/m$, then $z^* > 0$,

where z^* is any optimal solution to problem (1), when x_0 is fixed.

In Table 2, we report the results provided by Algorithm $FC_{\textcircled{1}}$ and expressed in terms of average testing correctness. We compare them with those ones relative to the two following fixed-center classical variants, obtained by setting

$$x_0 = x_0^A \quad (\text{Algorithm } FC_A)$$

Table 2 Numerical results

Data set	$FC_{\mathcal{A}}$	$FC_{\mathcal{AB}}$	$FC_{\textcircled{1}}$	SVM_0
Cancer	97.00	95.71	<u>97.57</u>	71.86
Diagnostic	83.86	53.33	89.65	<u>92.11</u>
Heart	74.33	55.00	<u>87.33</u>	68.67
Pima	<u>69.35</u>	66.23	61.43	61.82
Ionosphere	51.14	40.75	<u>78.86</u>	69.43
Sonar	59.05	52.86	65.71	<u>75.24</u>
Mushrooms	76.44	64.50	<u>78.19</u>	49.59
Prognosis	56.00	45.00	<u>68.00</u>	53.00
Tic Tac Toe	<u>71.79</u>	70.42	57.79	50.11
Votes	82.79	53.35	<u>86.74</u>	76.51
Galaxy	80.24	51.36	<u>89.19</u>	54.32
g50c	67.62	50.26	<u>90.58</u>	86.56
g10n	53.58	45.02	77.66	<u>90.24</u>

Underlined values are the best results in terms of average testing correctness

and

$$x_0 = x_0^{\mathcal{A}} + x_0^{\mathcal{B}} \quad (\text{Algorithm } FC_{\mathcal{AB}}),$$

respectively, and with the results obtained by a variant of the standard linear SVM (Algorithm SVM_0), where, in order to have a fair comparison, we have dropped the margin term by setting, in the `fitsvm` MATLAB subroutine, the penalty parameter `BoxConstraint` equal to 10^6 . We recall in fact that our spherical approach does not involve any margin concept. In Table 2, for each data set, the best result is underlined.

In comparison with $FC_{\mathcal{A}}$ and $FC_{\mathcal{AB}}$, the choice of the infinitely far center appears to be the best one; in fact, Algorithm $FC_{\textcircled{1}}$ outperforms the other two approaches on all the data sets except Pima and Tic Tac Toe, where the best performance is got by fixing x_0 as the barycenter of \mathcal{A} . We note also that choosing x_0 as the barycenter of all the points is not a good strategy, since the corresponding results are very poor on all the test problems, but Cancer and Tic Tac Toe, where the testing correctnesses appear comparable.

Also with respect to SVM_0 , Algorithm $FC_{\textcircled{1}}$ is characterized by a good performance, except on Diagnostic, Sonar and g10n, while on Pima both the approaches behave almost the same. These results were expected because, even if taking the radius infinitely far makes the spherical separability tend to the linear separability, the two approaches differ substantially. We recall in fact that if two sets are linearly separable, they are also spherical separable (even taking a very large radius), but the vice versa is not true.

5 Conclusions

In this paper, we have launched the idea to use the grossone theory within classification problems. In particular, we have focused our attention on the possibility to construct a binary spherical classifier characterized by an infinitely far center. As shown by the numerical results, adopting the grossone theory allows to obtain a good performance in terms of average testing correctness, managing very easily the numerical computations, which do not require any tuning of the “big M ” parameter.

Future research could consist in extending such approach to the kernel trick, which is well suitable in the fixed-center spherical separation, as shown in Astorino and Gaudio (2009), and to introduce the margin concept as in Astorino et al. (2012b).

Acknowledgements No funding was received.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

- Andrews S, Tsochantaridis I, Hofmann T (2003) Support vector machines for multiple-instance learning. In: Becker S, Thrun S, Obermayer K (eds) *Advances in neural information processing systems*. MIT Press, Cambridge, pp 561–568
- Astorino A, Fuduli A (2007) Nonsmooth optimization techniques for semisupervised classification. *IEEE Trans Pattern Anal Mach Intell* 29(12):2135–2142
- Astorino A, Fuduli A (2016) The proximal trajectory algorithm in SVM cross validation. *IEEE Trans Neural Netw Learn Syst* 27(5):966–977
- Astorino A, Gaudio M (2009) A fixed-center spherical separation algorithm with kernel transformations for classification problems. *Comput Manag Sci* 6(3):357–372
- Astorino A, Fuduli A, Gorgone E (2008) Non-smoothness in classification problems. *Optim Methods Softw* 23(5):675–688
- Astorino A, Fuduli A, Gaudio M (2010) DC models for spherical separation. *J Glob Optim* 48(4):657–669
- Astorino A, Bomze I, Brito P, Gaudio M (2012a) Two spherical separation procedures via non-smooth convex optimization. In: Simone VD, Serafino DD, Toraldo G (eds) *Recent advances in nonlinear optimization and equilibrium problems: a tribute to Marco D’Apuzzo*, *Quaderni di Matematica*, Dipartimento di Matematica della Seconda Università di Napoli, vol 27, pp 1–16. Aracne
- Astorino A, Fuduli A, Gaudio M (2012b) Margin maximization in spherical separation. *Comput Optim Appl* 53(2):301–322
- Astorino A, Gaudio M, Khalaf W (2014a) Edge detection by spherical separation. *Comput Manag Sci* 11(4):517–530

- Astorino A, Gaudio M, Seeger A (2014b) Conic separation of finite sets. I. The homogeneous case. *J Convex Anal* 21(1):1–28
- Astorino A, Gaudio M, Seeger A (2014c) Conic separation of finite sets. II. The nonhomogeneous case. *J Convex Anal* 21(3):819–831
- Astorino A, Fuduli A, Gaudio M (2016) Nonlinear programming for classification problems in machine learning. In: AIP conference proceedings, vol 1776, 040004
- Astorino A, Bomze I, Fuduli A, Gaudio M (2017) Robust spherical separation. *Optimization* 66(6):925–938
- Astorino A, Gaudio M, Fuduli A, Vocaturo E (2018) A multiple instance learning algorithm for color images classification. In: ACM international conference proceeding series, pp 262–266 . <https://doi.org/10.1145/3216122.3216144>
- Astorino A, Fuduli A, Gaudio M (2019a) A Lagrangian relaxation approach for binary multiple instance classification. *IEEE Trans Neural Netw Learn Syst* 30(9):2662–2671
- Astorino A, Fuduli A, Giallombardo G, Miglionico G (2019b) SVM-based multiple instance classification via DC optimization. *Algorithms* 12(12):249
- Astorino A, Fuduli A, Veltri P, Vocaturo E (2020) Melanoma detection by means of multiple instance learning. *Interdiscip Sci Comput Life Sci* 12(1):24–31
- Avolio M, Fuduli A (2020) A semiproximal support vector machine approach for binary multiple instance learning. *IEEE Transactions on Neural Networks and Learning Systems*. <https://doi.org/10.1109/TNNLS.2020.3015442> (in press)
- Caldarola F (2018) The Sierpinski curve viewed by numerical computations with infinities and infinitesimals. *Appl Math Comput* 318:321–328
- Celebi ME (ed) (2015) *Partitional clustering algorithms*. Springer, Berlin
- Chapelle O, Schölkopf B, Zien A (eds) (2006) *Semi-supervised learning*. MIT Press, Cambridge
- Chapelle O, Zien A (2005) Semi-supervised classification by low density separation. In: *Proceedings of the tenth international workshop on artificial intelligence and statistics*, pp 57–64
- Cococcioni M, Cudazzo A, Pappalardo M, Sergeyev Y.D (2020) Solving the lexicographic multi-objective mixed-integer linear programming problem using branch-and-bound and grossone methodology. *Communications in Nonlinear Science and Numerical Simulation* p. in Press . <https://doi.org/10.1016/j.cnsns.2020.105177>
- Cristianini N, Shawe-Taylor J (2000) *An introduction to support vector machines and other kernel-based learning methods*. Cambridge University Press, Cambridge
- De Cosmis S, De Leone R (2012) The use of grossone in mathematical programming and operations research. *Appl Math Comput* 218(16):8029–8038
- De Leone R (2018) Nonlinear programming and grossone: quadratic programming and the role of constraint qualifications. *Appl Math Comput* 318:290–297
- De Leone R, Fasano G, Sergeyev YD (2018) Planar methods and grossone for the conjugate gradient breakdown in nonlinear programming. *Comput Optim Appl* 71:73–93
- De Leone R, Fasano G, Roma M, Sergeyev Y (2020) Iterative grossone-based computation of negative curvature directions in large-scale optimization. *J Optim Theory Appl* 186(2):554–589
- Falcone A, Garro A, Mukhametzhano MS, Sergeyev YD (2020) A simulink-based infinity computer simulator and some applications. In: Sergeyev YD, Kvasov DE (eds) *Numerical computations: theory and algorithms*. Springer, Cham, pp 362–369
- Gaudio M, Giallombardo G, Mukhametzhano MS (2018) Numerical infinitesimals in a variable metric method for convex nonsmooth optimization. *Appl Math Comput* 318:312–320
- Gaudio M, Giallombardo G, Miglionico G, Vocaturo E (2020) Classification in the multiple instance learning framework via spherical separation. *Soft Comput* 24(7):5071–5077
- Herrera F, Ventura S, Bello R, Cornelis C, Zafra A, Sánchez-Tarragó D, Vluymans S (2016) *Multiple instance learning: foundations and algorithms*. Springer, Berlin
- Iavernaro F, Mazzia F, Mukhametzhano MS, Sergeyev YD (2020) Conjugate-symplecticity properties of Euler–Maclaurin methods and their implementation on the infinity computer. *Appl Numer Math*. <https://doi.org/10.1016/j.apnum.2019.06.011>
- Le Thi HA, Minh LH, Pham Dinh T, Ngai VH (2013) Binary classification via spherical separator by DC programming and DCA. *J Glob Optim* 56:1393–1407
- Lolli G (2015) Metamathematical investigations on the theory of grossone. *Appl Math Comput* 255:3–14
- Mangasarian OL (1965) Linear and nonlinear separation of patterns by linear programming. *Oper Res* 13(3):444–452
- Margenstern M (2011) Using grossone to count the number of elements of infinite sets and the connection with bijections. *p-Adic numbers. Ultrametric Anal Appl* 3(3):196–204
- Montagna F, Simi G, Sorbi A (2015) Taking the Pirahā seriously. *Commun Nonlinear Sci Numer Simul* 21(1–3):52–69
- Murphy PM, Aha DW (1992) UCI repository of machine learning databases. In: www.ics.uci.edu/~mllearn/MLRepository.html
- Odewahn S, Stockwell E, Pennington R, Humphreys R, Zumach W (1992) Automated star/galaxy discrimination with neural networks. *Astronom J* 103(1):318–331
- Plastria F, Carrizosa E, Gordillo J (2014) Multi-instance classification through spherical separation and VNS. *Comput Oper Res* 52:326–333
- Sergeyev YD (2003) *Arithmetic of Infinity*. Edizioni Orizzonti Meridionali, CS (2003, 2nd ed. 2013)
- Sergeyev YD (2009) Numerical point of view on calculus for functions assuming finite, infinite, and infinitesimal values over finite, infinite, and infinitesimal domains. *Nonlinear Anal Ser A Theory Methods Appl* 71(12):e1688–e1707
- Sergeyev YD (2010a) Computer system for storing infinite, infinitesimal, and finite quantities and executing arithmetical operations with them. USA patent 7,860,914
- Sergeyev YD (2010) Lagrange lecture: methodology of numerical computations with infinities and infinitesimals. *Rendiconti del Seminario Matematico dell'Università e del Politecnico di Torino* 68(2):95–113
- Sergeyev YD (2011) Higher order numerical differentiation on the infinity computer. *Optim Lett* 5(4):575–585
- Sergeyev YD (2015) Un semplice modo per trattare le grandezze infinite ed infinitesime. *Matematica nella Società e nella Cultura: Rivista della Unione Matematica Italiana* 8(1): 111–147
- Sergeyev YD (2017) Numerical infinities and infinitesimals: methodology, applications, and repercussions on two Hilbert problems. *EMS Surv Math Sci* 4:219–320
- Sergeyev YD (2018) Numerical infinities applied for studying Riemann series theorem and Ramanujan summation. In: AIP conference proceedings of ICNAAM 2017, vol 1978, p 020004. AIP Publishing, New York . <https://doi.org/10.1063/1.5043649>
- Sergeyev YD (2019) Independence of the grossone-based infinity methodology from non-standard analysis and comments upon logical fallacies in some texts asserting the opposite. *Found Sci* 24(1):153–170
- Sergeyev YD, Kvasov DE, Mukhametzhano MS (2018) On strong homogeneity of a class of global optimization algorithms working with infinite and infinitesimal scales. *Commun Nonlinear Sci Numer Simul* 59:319–330
- Tax DMJ, Duin RPW (1999) Data domain description using support vectors. In: *ESANN'1999 proceedings Bruges*, pp 251–256. Belgium

- Vapnik V (1995) The nature of the statistical learning theory. Springer, New York
- Zhigljavsky A (2012) Computing sums of conditionally convergent and divergent series using the concept of grossone. Appl Math Comput 218(16):8064–8076

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.