Soft Computing (2022) 26:3361–3368
https://doi.org/10.1007/s00500-021-06713-1

OPTIMIZATION

A heuristic approach for multiple instance learning by linear separation

Antonio Fuduli[1] · Manlio Gaudioso[2] · Walaa Khalaf[3] · Eugenio Vocaturo[2,4]

Accepted: 20 December 2021 / Published online: 17 January 2022

Abstract
We present a fast heuristic approach for solving a binary multiple instance learning (MIL) problem, which consists in discriminating between two kinds of item sets: the sets are called bags and the items inside them are called instances. Assuming that only two classes of instances are allowed, a common standard hypothesis states that a bag is positive if it contains at least a positive instance and it is negative when all its instances are negative. Our approach constructs a MIL separating hyperplane by preliminary fixing the normal and reducing the learning phase to a univariate nonsmooth optimization problem, which can be quickly solved by simply exploring the kink points. Numerical results are presented on a set of test problems drawn from the literature.

Keywords Multiple instance learning (MIL) · Linear separation · Nonsmooth optimization

1 Introduction

Multiple instance learning (MIL) (Herrera et al. 2016) is about classification of sets of items: in the MIL terminology, such sets are called *bags* and the corresponding items are called *instances*. In the binary case, when also the instances can belong only to two alternative classes, a MIL problem is stated on the basis of the so-called *standard MIL assumption*, which refers to a positive bag as a bag containing at least a positive instance and to a negative one as any bag whose instances are all negative. For this reason, MIL problems are often interpreted as a kind of weakly supervised classification problems.

The first MIL problem encountered in the literature is a drug design problem (Dietterich et al. 1997). It consists in determining whether a drug molecule is active or not. Each molecule can assume a finite number of three-dimensional conformations, and it is active if at least one among its conformations is able to bind a particular " binding site," which generally coincides with a larger protein molecule. The key question is that it is not known which conformation makes a molecule active. In this example, the drug molecule is represented by a bag, while various conformations it can assume correspond to the instances inside the bag.

The MIL paradigm finds application in a lot of fields: text categorization, image recognition (Astorino et al. 2017, 2018), video analysis, diagnostics by means of images (Astorino et al. 2019b, 2020, and Quellec et al. 2017) and so on. An example fitting very well the standard MIL assumption stated above is in discriminating between healthy and nonhealthy patients on the basis of their medical scan (bag): if at least a region (instance) of the medical scan (bag) is abnormal, then the patient is classified as nonhealthy and, on the contrary, when all the regions (instances) of the medical scan (bag) are normal, then the patient is classified as healthy.

In the last years, many papers have been devoted to MIL problems. Various approaches discussed in the literature

✉ Antonio Fuduli
antonio.fuduli@unical.it

Manlio Gaudioso
gaudioso@dimes.unical.it

Walaa Khalaf
w.khalaf@uomustansiriyah.edu.iq

Eugenio Vocaturo
e.vocaturo@dimes.unical.it

[1] Department of Mathematics and Computer Science, University of Calabria, 87036 Rende, CS, Italy

[2] Department of Computer Engineering, Modeling, Electronics and Systems, University of Calabria, 87036 Rende, CS, Italy

[3] Computer Engineering Department, College of Engineering-Mustansiriyah University, Baghdad 10047, Iraq

[4] Nanotec, Italian National Research Council, 87036 Rende (CS), Italy

fall into one of the three following classes: instance-space approaches, bag-space approaches and embedding-space approaches. In the instance-space approaches, the classifier is constructed in the instance space and the classification of the bags is inferred from the classification of the instances: as a consequence, this kind of approach is of the local type. Some instance-space approaches are found in Andrews et al. (2003); Astorino et al. (2019a, c); Avolio and Fuduli (2021); Bergeron et al. (2012); Gaudioso et al. (2020b); Mangasarian and Wild (2008); Vocaturo et al. (2020). In particular, in Andrews et al. (2003) the first SVM (Support Vector Machine) type model for MIL has been proposed, giving rise to a nonlinear mixed integer program solved by means of a BCD (Block Coordinate Descent) approach (Tseng 2001). The same SVM type model treated in Andrews et al. (2003) has been faced in Astorino et al. (2019a) by means of a Lagrangian relaxation technique, while in Astorino et al. (2019c) and Bergeron et al. (2012) a MIL linear separation has been obtained by using some ad hoc nonsmooth approaches. In Mangasarian and Wild (2008), the authors have proposed an instance-space algorithm, expressing each positive bag as convex combination of its instances, whereas in Avolio and Fuduli (2021) a combination of the SVM and the PSVM (Proximal Support Vector Machine) approaches has been adopted. In Gaudioso et al. (2020b) and Vocaturo et al. (2020), a spherical separation model has been tackled by using DC (Difference of Convex) techniques. Other SVM type instance space approaches for MIL are found in Li et al. (2009), Melki et al. (2018), Shan et al. (2018), and Zhang et al. (2013), while in Yuan et al. (2021) a spherical separation with margin is used.

Differently from the above instance-space approaches, the bag-space techniques (see for example Gärtner et al. (2002), Wang and Zucker (2000), and Zhou et al. (2009)) are of the global type since classification is performed considering each bag as an entire entity. Finally, the embedding-space approaches, such as Zhang et al. (2017), are a compromise between the two previous ones since the classifier is obtained in the instance space on the basis of some instances per bag, those ones, in particular, which are more representative of the bag. For more details on the MIL paradigm, we refer the reader to the exhaustive surveys Amores (2013) and Carbonneau et al. (2018).

In this work, stemming from a formulation similar to those adopted in Andrews et al. (2003) (MI-SVM formulation), Astorino et al. (2019c), and Bergeron et al. (2012), where both the normal and the bias of a separation hyperplane are computed, we present a fast instance-space algorithm, which generates a separation hyperplane by heuristically prefixing its normal and by successively computing the bias as an optimal solution to an univariate nonsmooth optimization problem. Solving efficiently this univariate nonsmooth problem (by simply exploring the kink points) constitutes the main novelty of our approach, which ensures quite low computational times while providing reasonable testing accuracy.

The paper is organized as follows. In Sect. 2, we introduce our approach, while some numerical results on a set of benchmark test problems are reported in Sect. 3. Finally, in Sect. 4 some conclusions are drawn.

2 The approach

Assume we are given the index sets I^- and I^+ of k negative and m positive bags, respectively. We indicate by $\{x_j \in \mathbb{R}^n\}$ the set of all the instances, each of them belonging to exactly one bag, either negative or positive. We assume $\{J_1^-, \ldots, J_k^-\}$ and $\{J_1^+, \ldots, J_m^+\}$ be the instance index sets of the negative and positive bags, respectively.

The objective is to find a hyperplane

$$H(w, \gamma) \stackrel{\triangle}{=} \{x \in \mathbb{R}^n | w^\top x = \gamma\},$$

with $w \in \mathbb{R}^n$ and $\gamma \in \mathbb{R}$, which (strictly) separates the two classes of bags on the basis of the standard MIL assumption, i.e.,

- all the negative bags are entirely confined in the interior of one of the two halfspaces generated by H;
- each positive bag has at least one of its instances falling into the interior of the other halfspace.

More formally, $H(w, \gamma)$ is a separating hyperplane if and only if:

$$\text{for each } i \in I^- \quad \text{it is } \ w^\top x_j \le \gamma - 1, \quad \text{for each } j \in J_i^-, \tag{1}$$

$$\text{for each } i \in I^+ \quad \text{it is } \ w^\top x_j \ge \gamma + 1, \quad \text{for at least one } \ j \in J_i^+. \tag{2}$$

To state an optimization model able to provide a possibly separating hyperplane, we define the error $e_i^-(w, \gamma)$ in classifying the negative bag $i \in I^-$ as

$$e_i^-(w, \gamma) \stackrel{\triangle}{=} \sum_{j \in J_i^-} \max \left\{0, w^\top x_j - \gamma + 1\right\}, \quad i = 1, \ldots, k,$$

and the error $e_i^+(w, \gamma)$ in classifying the positive bag J_i^+ as

$$e_i^+(w, \gamma) \stackrel{\triangle}{=} \max \left\{0, \min_{j \in J_i^+} \{-w^\top x_j + \gamma + 1\}\right\}, \quad i = 1, \ldots, m.$$

Summing up, we obtain the following overall error function:

$$e(w,\gamma) \triangleq \sum_{i\in I^-} e_i^-(w,\gamma) + \sum_{i\in I^+} e_i^+(w,\gamma) = \sum_{i\in I^-}\sum_{j\in J_i^-} \max\left\{0, w^\top x_j - \gamma + 1\right\} + \sum_{i\in I^+} \max\left\{0, \min_{j\in J_i^+}\{-w^\top x_j + \gamma + 1\}\right\} \tag{3}$$

and the resulting optimization problem

$$\min_{(w,\gamma)\in\mathbb{R}^{n+1}} e(w,\gamma). \tag{4}$$

We will refer to the model above as to *Formulation 1*. Note that $e(w,\gamma) \geq 0$ and it is $e(w,\gamma) = 0$ if and only if $H(w,\gamma)$ is a separating hyperplane, according to (1) and (2).

Function $e(w,\gamma)$ is nonsmooth and nonconvex, but it can be put in DC (Difference of Convex) form (Le Thi and Pham Dinh 2005). This formulation is similar to those adopted in Andrews et al. (2003) (MI-SVM formulation) and Bergeron et al. (2012), while the DC decomposition has been exploited in Astorino et al. (2019c). The reader will find a fresh survey on nonsmooth optimization methods in Gaudioso et al. (2020a). Some specialized algorithms can be found in Gaudioso and Monaco (1992) and Astorino et al. (2011).

Our heuristic approach consists first in a judicious selection of w, the normal to the separating hyperplane, and then in minimizing the error function with respect to the scalar variable γ.

As for the choice of w, we calculate the barycenter a of all the instances of the negative bags and the barycenter b of the barycenters of the instances in each positive bag, and then, we fix the normal to the hyperplane $\bar{w}$ by setting:

$$\bar{w} = M(b-a), \tag{5}$$

for some $M > 0$. Note that, whenever $M = 1$, provided a and b do not coincide, by setting $\gamma_- = a^\top b - \|a\|^2$ and $\gamma_+ = \|b\|^2 - a^\top b$, the hyperplanes $H(\bar{w},\gamma_-)$ and $H(\bar{w},\gamma_+)$ pass through points a and b, respectively.

Once the normal $\bar{w}$ has been fixed, defining

$$\alpha_{ij} \triangleq \bar{w}^\top x_j + 1, \quad i \in I^-,\ j \in J_i^- \tag{6}$$

and

$$\beta_i \triangleq \min_{j\in J_i^+}\{-\bar{w}^\top x_j + 1\}, \quad i \in I^+, \tag{7}$$

we rewrite function (3) as follows:

$$e(\gamma) = \sum_{i\in I^-} e_i^-(\bar{w},\gamma) + \sum_{i\in I^+} e_i^+(\bar{w},\gamma) = \sum_{i\in I^-}\sum_{j\in J_i^-} \max\{0, \alpha_{ij} - \gamma\} + \sum_{i\in I^+} \max\{0, \beta_i + \gamma\}. \tag{8}$$

As a consequence, problem (4) becomes

$$\min_{\gamma\in\mathbb{R}} e(\gamma), \tag{9}$$

which consists of minimizing a convex and nonsmooth (piecewise affine) function of the scalar variable γ.

We note in passing that, by introducing the additional variables ξ_{ij}, $i \in I^-$, $j \in J_i^-$, and ζ_i, $i \in I^+$ (grouped into the vectors ξ, ζ), the problem can be equivalently rewritten as a linear program of the form

$$\begin{cases} \min\limits_{\xi,\zeta,\gamma} \sum\limits_{i\in I^-}\sum\limits_{j\in J_i^-} \xi_{ij} + \sum\limits_{i\in I^+} \zeta_i \\ \quad \xi_{ij} \geq \alpha_{ij} - \gamma, & i \in I^-,\ j \in J_i^- \\ \quad \zeta_i \geq \beta_i + \gamma, & i \in I^+ \\ \quad \xi_{ij} \geq 0, & i \in I^-,\ j \in J_i^- \\ \quad \zeta_i \geq 0, & i \in I^+. \end{cases}$$

To find an optimal solution to the problem, we prefer, however, to consider formulation (9). Note that the nonnegative function $e(\gamma)$ is continuous and coercive; consequently, it has a minimum. In particular,

$$e^* \triangleq \min_{\gamma\in\mathbb{R}} e(\gamma) = 0$$

corresponds to a correct classification of all the bags.

A brief discussion of the differential properties of function (8) is in order. Letting

$$r_{ij}(\gamma) \triangleq \max\{0, \alpha_{ij} - \gamma\}, \quad i \in I^-,\ j \in J_i^-$$

and

$$q_i(\gamma) \triangleq \max\{0, \beta_i + \gamma\}, \quad i = 1,\ldots,m,$$

we have the following expressions of the correspondent subdifferentials:

$$\partial r_{ij}(\gamma) = \begin{cases} \{-1\} & \text{if } \gamma < \alpha_{ij} \\ [-1,0] & \text{if } \gamma = \alpha_{ij} \\ \{0\} & \text{if } \gamma > \alpha_{ij} \end{cases} \tag{10}$$

and

$$\partial q_i(\gamma) = \begin{cases} \{0\} & \text{if } \gamma < -\beta_i \\ [0,\,1] & \text{if } \gamma = -\beta_i \\ \{1\} & \text{if } \gamma > -\beta_i \end{cases} \tag{11}$$

From (10) and (11), it is easy to see that the points α_{ij}, $i \in I^-$, $j \in J_i^-$, and $-\beta_i$, $i \in I^+$, constitute the *kinks*, i.e., the points where function $e(\gamma)$ is nonsmooth. Note also that $e(\gamma)$ has constant negative slope

$$-\sum_{i \in I^-} |J_i^-|$$

for

$$\gamma < \min\left\{ \min_{i \in I^-,\, j \in J_i^-} \alpha_{ij},\ \min_{i \in I^+} -\beta_i \right\},$$

and it has constant positive slope $m = |I^+|$ for

$$\gamma > \max\left\{ \max_{i \in I^-,\, j \in J_i^-} \alpha_{ij},\ \max_{i \in I^+} -\beta_i \right\}.$$

Taking into account (10) and (11), the subdifferential $\partial e(\gamma)$ is the Minkowski sum of four sets, i.e.,

$$\begin{aligned} \partial e(\gamma) = \big\{ -|IJ^-(\gamma)| \big\} + \big\{ |I^+(\gamma)| \big\} + [-1,\,0]|IJ_0^-(\gamma)| \\ +[0,\,1]|I_0^+(\gamma)|, \end{aligned} \tag{12}$$

with

$$\begin{aligned} IJ^-(\gamma) &= \big\{(i,\,j)\ |i \in I^-,\, j \in J_i^-,\ \gamma < \alpha_{ij}\big\}, \\ I^+(\gamma) &= \big\{i \mid i \in I^+,\ \gamma > -\beta_i\big\}, \\ IJ_0^-(\gamma) &= \big\{(i,\,j)\ |i \in I^-,\, j \in J_i^-,\ \gamma = \alpha_{ij}\big\}, \\ I_0^+(\gamma) &= \big\{i \mid i \in I^+,\ \gamma = -\beta_i\big\}, \end{aligned}$$

and, at the non-kinks points where function $e(\gamma)$ is differentiable, it is

$$e'(\gamma) = -|IJ^-(\gamma)| + |I^+(\gamma)|. \tag{13}$$

Moreover, at each kink point, say γ, the slope jumps up of s, the multiplicity of the kink defined as $|IJ_0^-(\gamma)| + |I_0^+(\gamma)|$.

Letting

$$\gamma_\alpha \stackrel{\triangle}{=} \max_{i \in I^-,\, j \in J_i^-} \alpha_{ij}$$

and

$$\gamma_\beta \stackrel{\triangle}{=} \min_{1 \le i \le m} -\beta_i,$$

the following property holds.

Proposition 1 *The optimal objective function value e^* of problem (9) is equal to zero if and only if*

$$\gamma_\alpha \le \gamma_\beta. \tag{14}$$

In such case, every $\gamma \in [\gamma_\alpha, \gamma_\beta]$ is optimal.

Proof Straightforward from (8). □

We consider now the case $\gamma_\alpha > \gamma_\beta$ and state the following theorem.

Theorem 1 *If $\gamma_\alpha > \gamma_\beta$, then there exists an optimal kink solution $\gamma^* \in [\gamma_\beta, \gamma_\alpha]$.*

Proof We prove first that any optimal solution belongs to the interval $[\gamma_\beta, \gamma_\alpha]$. We observe in fact that, for every $\bar\gamma < \gamma_\beta$ and for $i \in I^+$, it is $\max\{0, \beta_i + \bar\gamma\} = 0$, while there exists at least one couple ij such that $\max\{0, \alpha_{ij} - \bar\gamma\} > 0$. This implies that the directional derivative of $e(\gamma)$ at $\bar\gamma$ along the positive semi-axis is negative. A similar argument can be used to show that at any $\bar\gamma > \gamma_\alpha$ the directional derivative along the positive semi-axis is positive. As a consequence, $e(\gamma)$ has optimal solution necessarily in the interval $[\gamma_\beta, \gamma_\alpha]$.

Now, observing that γ_α and γ_β are both kinks, consider any optimal non-kink solution $\gamma^* \in (\gamma_\beta, \gamma_\alpha)$. Since the function is differentiable at γ^*, it follows that the derivative of $e(\gamma)$ at γ^* vanishes, that is, from (13),

$$-|IJ^-(\gamma^*)| + |I^+(\gamma^*)| = 0,$$

i.e., $|IJ_0^-(\gamma^*)| = |I_0^+(\gamma^*)| = 0$.

Now consider the biggest kink smaller than γ^*: the existence of such a kink is guaranteed recalling that γ_β is a kink and $\gamma^* > \gamma_\beta$. Assume for the time being that such a kink is α_{sh} for some $s \in I^-$, $h \in J_s^-$ and let $\bar\gamma = \alpha_{sh}$. It is

$$\begin{aligned} IJ^-(\bar\gamma) &= IJ^-(\gamma^*) \\ I^+(\bar\gamma) &= I^+(\gamma^*) \\ |IJ_0^-(\bar\gamma)| &= 1 \\ |I_0^+(\bar\gamma)| &= 0 \end{aligned}$$

Summing up and taking into account (12), it follows $0 \in \partial e(\bar\gamma)$, i.e., $\bar\gamma = \alpha_s$ is an optimal kink solution. The case when the biggest kink smaller than γ^* is $-\beta_s$ for some $s \in I^+$ can be treated in a perfectly analogous way. □

The properties of function $e(\gamma)$ we have discussed allow us to state the following kink exploring algorithm to solve problem (9) in order to compute an optimal solution γ^*.

Algorithm MIL-kink

Step 0 (Computing the kinks). Given $\bar w$, compute the kinks α_{ij}, $i \in I^-$, $j \in J_i^-$ and β_i, $i \in I^+$. Compute γ_α and γ_β. If $\gamma_\alpha \le \gamma_\beta$, STOP: choose γ^* as any value in the interval $[\gamma_\alpha, \gamma_\beta]$.

Table 1 Data sets

Data set	Dimension (n)	Instances	Bags ($m+k$)	Positive bags (m)	Negative bags (k)
Elephant	230	1391	200	100	100
Fox	230	1320	200	100	100
Tiger	230	1220	200	100	100
TST1	6668	3224	400	200	200
TST2	6842	3344	400	200	200
TST3	6568	3246	400	200	200
TST4	6626	3391	400	200	200
TST7	7037	3367	400	200	200
TST9	6982	3300	400	200	200
TST10	7073	3453	400	200	200
Musk-1	166	476	92	47	45
Musk-2	166	6598	102	39	63

Step 1 (Sorting the kinks). Order the kinks in the interval $[\gamma_\beta, \gamma_\alpha]$ for increasing values of the α_{ij}'s and of the $-\beta_i$'s.

Step 2 (Exploring the kinks). Explore the kinks starting from γ_β until a value $\gamma^* = \alpha_{sh}$ for some $s \in I^-$, $h \in J_s^-$ or $\gamma^* = \beta_s$ for some $s \in I^+$ is found such that $0 \in \partial e(\gamma^*)$.

Proposition 2 *Algorithm MIL-kink runs in time $O(p)$, where*

$$p \stackrel{\triangle}{=} \max\{n\bar{k}, n\bar{m}, (m+\bar{k})\log(m+\bar{k})\},$$

with $\bar{m}$ and $\bar{k}$ being the total number of instances in the positive and negative bags, respectively.

Proof The computation of the α_{ij}s and β_is at Step 0 is performed in time $O(n\bar{k})$ and $O(n\bar{m})$, respectively, while the computation of γ_α and γ_β takes time $O(\bar{k})$ and $O(m)$, respectively. Sorting the kinks at Step 1 takes time $O((m+\bar{k})\log(m+\bar{k}))$, while exploring the kinks at Step 2 is performed in time $O(m+\bar{k})$. The thesis follows. □

We conclude this section by remarking that an alternative formulation of the error function is obtained by replacing function $e_i^-(w,\gamma)$ in (3) by

$$e_i^-(w,\gamma) \stackrel{\triangle}{=} \max\left\{0, \max_{j\in J_i^-}\{w^\top x_j - \gamma + 1\}\right\}.$$

Such formulation will be referred to as *Formulation 2* and its theoretical treatment is perfectly analogous to that one of *Formulation 1*. Despite that, the two formulations present a relevant difference from the computational point of view, since in Formulation 2 the kinks α_{ij}, $i \in I^-$, $j \in J_i^-$, characterizing Formulation 1 (see formula 6), are replaced by the following ones:

$$\alpha_i \stackrel{\triangle}{=} \max_{j\in J_i^-}\{\bar{w}^\top x_j + 1\}, \quad i \in I^-,$$

Table 2 Numerical results: average training and testing correctnesses

Data set	MIL-kink[1] Training %	Testing %	MIL-kink[2] Training %	Testing %
Elephant	87.11	84.50	86.06	84.50
Fox	71.17	56.00	71.61	56.50
Tiger	75.72	72.50	81.78	79.00
TST1	94.94	92.75	95.00	92.25
TST2	76.25	70.75	77.17	72.00
TST3	84.03	78.50	84.22	79.00
TST4	85.58	80.25	85.97	81.50
TST7	87.75	82.50	88.64	82.50
TST9	68.33	65.25	72.61	68.50
TST10	88.33	82.75	90.72	84.75
Musk-1	73.73	66.67	74.94	70.00
Musk-2	65.87	67.00	63.91	59.00

which are much less. As a consequence,

$$\gamma_\alpha \stackrel{\triangle}{=} \max_{i\in I^-} \alpha_i.$$

In such case, Algorithm MIL-kink runs in time $O(q)$, where

$$q \stackrel{\triangle}{=} \max\{n\bar{m}, n\bar{k}, (m+k)\log(m+k)\}.$$

3 Numerical results

Algorithm MIL-kink, described in the previous section, has been implemented in MATLAB (version R2017b) on a Windows 10 system, characterized by a 2.21 GHz processor and 16 GB of RAM. Both the formulations (code MIL-kink[1], corresponding to Formulation 1, and code MIL-kink[2], corresponding to Formulation 2, have been tested on twelve data

Table 3 Numerical results: comparisons in terms of average testing correctness and average CPU time

Data set	MIL-kink[1]		MIL-kink[2]		mi-SPSVM		mi-SVM		MIL-RL	
	%	secs	%	secs	%	secs	%	secs	%	secs
Elephant	84.50	0.01	84.50	0.00	76.50	1.36	84.00	8.97	82.50	11.75
Fox	56.00	0.01	56.50	0.00	59.00	2.78	54.00	14.74	53.00	16.77
Tiger	72.50	0.00	79.00	0.00	74.50	1.59	77.00	6.43	75.00	9.20
TST1	92.75	0.24	92.25	0.16	94.25	5.49	94.75	103.10	95.25	146.49
TST2	70.75	0.28	72.00	0.16	74.50	6.09	86.25	128.30	86.25	192.53
TST3	78.50	0.26	79.00	0.16	86.25	5.62	81.00	81.70	86.75	185.71
TST4	80.25	0.27	81.50	0.17	81.75	6.12	79.00	174.85	69.50	185.92
TST7	82.50	0.31	82.50	0.15	81.75	6.05	79.00	170.81	71.25	178.72
TST9	65.25	0.28	68.50	0.14	68.50	6.05	60.75	37.95	69.00	110.51
TST10	82.75	0.29	84.75	0.16	78.75	7.98	71.75	150.31	80.50	186.42
Musk-1	66.67	0.00	70.00	0.00	82.22	0.08	76.67	0.27	72.22	0.62
Musk-2	67.00	0.04	59.00	0.01	73.00	295.34	68.00	437.64	73.00	963.64

sets drawn from the literature (Andrews et al. 2003) and are listed in Table 1. The first three data sets are image recognition problems, the last two ones consist in predicting whether a compound is a musk or not, while the TST data sets are large-scale text classification problems.

In all the experimentations, we have set $\bar{w}$ according to (5), taking $M = 10^6$. Moreover, for each data set, we have adopted the classical tenfold cross-validation, coming out with the results reported in Table 2 in terms of average training and testing correctness.

In Table 3, we compare our results, in terms of average testing correctness and average CPU time, with those ones reported in Avolio and Fuduli (2021) and provided by the MATLAB implementations (launched on the same machine, with the same cross-validation fold structure) of the following algorithms taken from the literature:

- mi-SPSVM (Avolio and Fuduli 2021): it is an instance-space approach, generating a separation hyperplane placed in the middle between a supporting hyperplane for the instances of the negative bags and a clustering hyperplane for the instances of the positive bags.
- mi-SVM (Andrews et al. 2003): it is an instance-space approach, where a separating hyperplane is constructed by solving an SVM type optimization model by means of a BCD technique Tseng (2001).
- MIL-RL (Astorino et al. 2019a): it is an instance-space approach, which provides a separating hyperplane by solving, by means of a Lagrangian relaxation technique Gaudioso (2020), the same SVM type optimization model adopted in mi-SVM.

All the above listed algorithms share with MIL-kink the characteristic of providing a linear separation classifier (i.e., a hyperplane); thus, the CPU time reported in Table 3 corresponds exactly to the execution time, averaged on tenfolds, needed to compute each time such a hyperplane.

In Table 3, for each data set, the best results have been underlined. Comparing all the algorithms, we observe that our approach is clearly very fast (with a CPU time always less than one second), especially when Formulation 2 is adopted (here the number of explored kinks is definitely smaller than in Formulation 1). In terms of average testing correctness, MIL-kink overcomes the other algorithms on four data sets (Elephant, Tiger, TST7, TST10), showing a comparable performance on the remaining test problems (especially on TST4 and TST9).

To have an idea of the performance of our method with respect to further approaches drawn from the literature, in Table 4 we report the comparison of our technique (only in terms of average testing correctness) against the following MIL algorithms, whose results have been taken from the corresponding papers:

- MI-NPSVM (Zhang et al. 2013): it is an instance-space approach, generating two nonparallel hyperplanes by solving two respective SVM type problems.
- MIRSVM (Melki et al. 2018): it is an embedding-space SVM type approach, based on identifying, at each iteration, the instances that mostly impact on the classification process.
- SSLM-MIL (Yuan et al. 2021): it is an instance space approach, based on spherical separation with margin.

For each data set, the best results have been underlined and the character " - " means that the corresponding datum is not available. Moreover, about mi-NPSVM, in order to have a fair comparison, we have considered only the linear kernel version, for which there is no result on the musk data sets.

Table 4 Numerical results: comparisons in terms of average testing correctness

Data set	MIL-kink[1] %	MIL-kink[2] %	MI-NPSVM %	MIRSVM %	SSLM-MIL %
Elephant	84.50	84.50	83.60	83.00	87.50
Fox	56.00	56.50	59.80	65.50	67.00
Tiger	72.50	79.00	82.10	77.50	84.50
TST1	92.75	92.25	92.80	–	–
TST2	70.75	72.00	85.90	–	–
TST3	78.50	79.00	84.80	–	–
TST4	80.25	81.50	84.10	–	–
TST7	82.50	82.50	81.50	–	–
TST9	65.25	68.50	67.80	–	–
TST10	82.75	84.75	80.50	–	–
Musk-1	66.67	70.00	–	90.22	91.40
Musk-2	67.00	59.00	–	82.18	87.20

Looking at the results of Table 4, we observe that MIL-kink exhibits the best performance on three data sets (TST7, TST9 and TST10) and it provides quite reasonable results also on Elephant, TST1 and TST4.

4 Conclusions

We have presented a fast heuristic algorithm for solving binary MIL problems characterized by two classes of instances. Our approach gives rise to a nonsmooth univariate optimization model that we solve exactly by simply exploring the kink points. The numerical results appear interesting mainly in terms of computational time, thus suggesting the use of the method either for dealing with very large data sets or as a first tool to check viability of a MIL approach in a specific application.

Author Contributions The authors contributed to each part of this paper equally.

Funding No funding was received.

Declarations

Conflict of interest All authors declare that he has 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

Amores J (2013) Multiple instance classification: review, taxonomy and comparative study. Artif Intell 201:81–105

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, Frangioni A, Gaudioso M, Gorgone E (2011) Piecewise quadratic approximations in convex numerical optimization. SIAM J Optim 21(4):1418–1438

Astorino A, Fuduli A, Gaudioso 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, Gaudioso M, Vocaturo E (2019b) Multiple instance learning algorithm for medical image classification. In: CEUR workshop proceedings, vol. 2400

Astorino A, Fuduli A, Giallombardo G, Miglionico G (2019c) SVM-based multiple instance classification via DC optimization. Algorithms 12:249

Astorino A, Fuduli A, Veltri P, Vocaturo E (2017) On a recent algorithm for multiple instance learning. preliminary applications in image classification. In: Proceedings - 2017 IEEE international conference on bioinformatics and biomedicine, BIBM 2017, vol 2017, pp 1615–1619

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

Astorino A, Gaudioso M, Fuduli A, Vocaturo E (2018) A multiple instance learning algorithm for color images classification. In: ACM international conference proceeding series, pp. 262–266

Avolio M, Fuduli A (2021) A semiproximal support vector machine approach for binary multiple instance learning. IEEE Trans Neural Netw Learn Syst 32(8):3566–3577

Bergeron C, Moore G, Zaretzki J, Breneman C, Bennett K (2012) Fast bundle algorithm for multiple instance learning. IEEE Trans Pattern Anal Mach Intell 34(6):1068–1079

Carbonneau M, Cheplygina V, Granger E, Gagnon G (2018) Multiple instance learning: a survey of problem characteristics and applications. Pattern Recogn 77:329–353

Dietterich T, Lathrop R, Lozano-Pérez T (1997) Solving the multiple instance problem with axis-parallel rectangles. Artif Intell 89(1–2):31–71

Gärtner T, Flach P, Kowalczyk A, Smola A (2002) Multi-instance kernels. In: In Proceedings of the 19th international conference on machine learning, pp. 179–186. Morgan Kaufmann

Gaudioso M (2020) A view of Lagrangian relaxation and its applications. Numerical nonsmooth optimization: state of the art algorithms. Springer, pp 579–617

Gaudioso M, Giallombardo G, Miglionico G (2020a) Essentials of numerical nonsmooth optimization. 4OR 18(1): 1–47

Gaudioso M, Giallombardo G, Miglionico G, Vocaturo E (2020b) Classification in the multiple instance learning framework via spherical separation. Soft Comput 24(7):5071–5077

Gaudioso M, Monaco M (1992) Variants to the cutting plane approach for convex nondifferentiable optimization. Optimization 25(1):65–75

Herrera F, Ventura S, Bello R, Cornelis C, Zafra A, Sánchez-Tarragó D, Vluymans S (2016) Multiple instance learning: foundations and algorithms. Springer International Publishing, Berlin

Le Thi H, Pham Dinh T (2005) The DC (difference of convex functions) programming and dca revisited with dc models of real world nonconvex optimization problems. J Global Optim 133:23–46

Li Y, Kwok JT, Tsang IW, Zhou Z (2009) A convex method for locating regions of interest with multi-instance learning, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol 5782 LNAI, pp 15–30

Mangasarian O, Wild E (2008) Multiple instance classification via successive linear programming. J Optim Theory Appl 137(3):555–568

Melki G, Cano A, Ventura S (2018) Mirsvm: multi-instance support vector machine with bag representatives. Pattern Recogn 79:228–241

Quellec G, Cazuguel G, Cochener B, Lamard M (2017) Multiple-instance learning for medical image and video analysis. IEEE Rev Biomed Eng 10:213–234

Shan C, Liu L, Xue J, Sun Z, Ma T (2018) Multiple-instance support vector machine based on a new local feature of hierarchical weighted spatio-temporal interest points. J Internet Technol 19(3)

Tseng P (2001) Convergence of a block coordinate descent method for nondifferentiable minimization. J Optim Theory Appl 109(3):475–494

Vocaturo E, Zumpano E, Giallombardo G, Miglionico G (2020) DC-SMIL: a multiple instance learning solution via spherical separation for automated detection of displastyc nevi. In: ACM international conference proceeding series

Wang J, Zucker JD (2000) Solving the multiple-instance problem: a lazy learning approach. In: Proceedings of the seventeenth international conference on machine learning, ICML '00, pp 1119–1126

Yuan M, Xu Y, Feng R, Liu Z (2021) Instance elimination strategy for non-convex multiple-instance learning using sparse positive bags. Neural Netw 142:509–521

Zhang Q, Perra N, Perrotta D, Tizzoni M, Paolotti D, Vespignani A (2017) Forecasting seasonal influenza fusing digital indicators and a mechanistic disease model. In: Proceedings of the 26th international conference on world wide web, WWW '17, pp 311–319

Zhang Q, Tian Y, Liu D (2013) Nonparallel support vector machines for multiple-instance learning. Procedia Comput Sci 17:1063–1072

Zhou ZH, Sun YY, Li YF (2009) Multi-instance learning by treating instances as non-IID samples. In: Proceedings of the 26th annual international conference on machine learning, pp 1249–1256. ACM