Machine Learning
https://doi.org/10.1007/s10994-023-06434-6

Online binary classification from similar and dissimilar data

Senlin Shu[1] · Haobo Wang[2] · Zhuowei Wang[3] · Bo Han[4] · Tao Xiang[1] · Bo An[5] · Lei Feng[5]

Received: 2 June 2023 / Revised: 11 August 2023 / Accepted: 7 October 2023

Abstract
Similar-dissimilar (SD) classification aims to train a binary classifier from only *similar* and *dissimilar* data pairs, which indicate whether two instances belong to the same class (similar) or not (dissimilar). Although effective learning methods have been proposed for SD classification, they cannot deal with online learning scenarios with sequential data that can be frequently encountered in real-world applications. In this paper, we provide the first attempt to investigate the *online SD classification* problem. Specifically, we first adapt the unbiased risk estimator of SD classification to online learning scenarios with a conservative regularization term, which could serve as a naive method to solve the online SD classification problem. Then, by further introducing a margin criterion for whether to update the classifier or not with the received cost, we propose two improvements (one with *linearly* scaled cost and the other with *quadratically* scaled cost) that result in two online SD classification methods. Theoretically, we derive the regret, mistake, and relative loss bounds for our proposed methods, which guarantee the performance on sequential data. Extensive experiments on various datasets validate the effectiveness of our proposed methods.

Keywords Similar-dissimilar classification · Unbiased risk estimator · Passive-aggressive method · Online learning

1 Introduction

Ordinary binary classification usually requires a vast amount of fully labeled data for training an accurate classifier. However, such large-scale fully labeled data may not be always available in some real-world scenarios due to the privacy, confidentiality, or security reasons. To alleviate this issue, many researchers have investigated various weakly supervised binary classification tasks, such as positive-unlabeled learning (Plessis et al., 2015; Cao et al., 2022), positive-confidence classification (Ishida et al., 2018; Shinoda et al., 2020), partial-label learning (Feng et al., 2020; Wu et al., 2022), similar-unlabeled learning (Bao et al., 2018), unlabeled-unlabeled learning (Lu et al., 2019, 2020), and similarity-confidence learning (Cao et al., 2021).

Editors: Vu Nguyen and Dana Yogatama.

Extended author information available on the last page of the article

This paper considers another weakly supervised binary classification task called *similar-dissimilar* (SD) classification (Shimada et al., 2020), which aims to train a binary classifier from only similar and dissimilar data pairs that indicate whether two instances belong to the same class (similar) or not (dissimilar). Compared with fully labeled data, similar and dissimilar data pairs could be easier to collect (Bao et al., 2018; Shimada et al., 2020). For example, it would be difficult to directly predict people's private or sensitive matters (e.g., religion and politics) because people may hesitate to give explicit answers to these matters. However, it could be easier for people to provide an answer to the question "which person do you have the same belief as". Additionally, in identifying proteins (Tao et al., 2004), it could be difficult for human to accurately distinguish identify whether a protein belongs to a certain protein super-family due to complex and diverse amino acid sequences. Fortunately, it could be easy to distinguish whether two proteins are similar or dissimilar since the similarity of protein structure could demonstrates the similarity of proteins. It is worth noting that although semi-supervised clustering (Li and Liu, 2009; Chen et al., 2022) can deal with such pairwise similarity information, there is a significant difference between *clustering* (Xu and Tian, 2015) and *classification* (Lu and Weng, 2007), and we focus on binary classification in this paper.

For binary classification with pairwise similarity information, effective empirical risk minimization methods have been proposed (Shimada et al., 2020; Bao et al., 2020). Although these methods have achieved satisfactory performance, all of them work in a batch learning mode, which cannot handle online learning scenarios with sequential data. In reality, we can frequently encounter that the collected pairwise data is presented in sequence for massive practical applications. Fortunately, *online learning* (Crammer and Singer, 2003; Crammer et al., 2009), which learns with training examples arriving in sequential order, has been extensively studied. In online learning, the classifier will be updated incrementally after receiving new training examples, so that the online learning methods do not need to require all training examples for training an accurate classifier. However, there still remains an open problem that has never been studied: *how can we train an effective binary classifier from only similar and dissimilar data coming in the online learning scenario?*

In this paper, we provide the first attempt to investigate *online binary classification from only similar and dissimilar data* (online SD classification for short). Our main contributions can be summarized as follows:

- We adapt the unbiased risk estimator of SD classification to online learning scenarios with a conservative regularizer, resulting in an online gradient descent algorithm.
- By further introducing a margin criterion for whether to update the classifier or not with the received cost, we propose two online passive-aggressive methods (with linearly scaled cost and quadratically scaled cost).
- We theoretically analyze the regret, mistake, and relative loss bounds for our proposed methods, which guarantee the performance on sequential data.

2 Related work

2.1 Similar-dissimilar classification

Binary classification with pairwise information is a weakly supervised learning problem, which has attracted increasing attention in recent years. It was shown (Bao et al., 2018) that

a binary classifier can be successfully learned from similar data pairs (i.e., two instances belonging to the same class) and unlabeled data by estimating the classification risk in an unbiased manner (the resulting estimator is called *unbiased risk estimator*). Later, *similar-dissimilar* (SD) classification (Shimada et al., 2020), which trains an effective binary classifier from only similar and dissimilar data pairs, was also investigated by empirical risk minimization with an unbiased risk estimator.

Here, we formally define the problem of SD classification and introduce the pioneer work (Shimada et al., 2020) that is most related to ours. Let $\mathcal{X} \in \mathbb{R}^d$ and $\mathcal{Y} = \{+1, -1\}$ be a d-dimensional feature space and binary label space, respectively. Suppose the collected dataset is represented as $\{(\boldsymbol{x}_i, \boldsymbol{x}'_i, s_i)\}_{i=1}^{n}$ where $s_i = +1$ indicates that $\boldsymbol{x}_i$ and $\boldsymbol{x}'_i$ belong to the same class (i.e., $y_i = y'_i$) and $s_i = -1$ indicates that $\boldsymbol{x}_i$ and $\boldsymbol{x}'_i$ belong to different classes (i.e., $y_i \neq y'_i$). With such a dataset, SD classification aims to train an effective binary classifier $f : \mathcal{X} \mapsto \mathbb{R}$ that tries to accurately predict the label of any unseen example. The pioneer study (Shimada et al., 2020) works by minimizing the following unbiased risk estimator:

$$\widehat{R}_{\mathrm{SD}}(f) = \frac{\pi_{\mathrm{S}}}{n_{\mathrm{S}}} \sum_{i=1}^{n_{\mathrm{S}}} \left[\frac{L(f(\boldsymbol{x}_i)) + L(f(\boldsymbol{x}'_i))}{2} \right] + \frac{\pi_{\mathrm{D}}}{n_{\mathrm{D}}} \sum_{i=1}^{n_{\mathrm{D}}} \left[\frac{L(-f(\boldsymbol{x}_i)) + L(-f(\boldsymbol{x}'_i))}{2} \right], \tag{1}$$

where $L(z) = \frac{\pi_+}{\pi_+ - \pi_-} \ell(z) - \frac{\pi_-}{\pi_+ - \pi_-} \ell(-z)$, $\ell(\cdot)$ denotes a binary loss (e.g., the hinge loss), π_+ (π_-) denotes the prior probability of the positive (negative) class, n_{S} (n_{D}) denotes the number of similar (dissimilar) data pairs and $\pi_{\mathrm{S}} = \pi_+^2 + \pi_-^2$ ($\pi_{\mathrm{D}} = 2\pi_+\pi_-$) denotes the fraction of similar (dissimilar) data pairs. Hence the total number of data pairs n equals to $n_{\mathrm{S}} + n_{\mathrm{D}}$, and n_{S} (n_{D}) can be calculated by n_{S}/n (n_{D}/n).

There are also other studies on SD classification by using a surrogate risk estimator (Bao et al., 2020) or dealing with noisy similarities (Maheshwara and Manwani, 2023). All existing methods work in a batch or offline learning mode, which cannot be used in online learning with sequential data.

2.2 Online learning

Online learning (Crammer and Singer, 2003; Kivinen et al., 2004; Hoi et al., 2021) aims to learn an incrementally updated model from data arriving in sequential order. Here, we introduce some supervised and weakly supervised online learning methods.

Supervised online learning methods update the classifier with fully supervised information. Popular supervised online learning methods include the perceptron-based algorithm (Freund and Schapire, 1999), the conformal prediction approach to the online binary classification with reject option (Koçak et al., 2016), the online universal classifier (Er et al., 2016), the online active learning algorithm (Liu et al., 2015), the distributed online algorithm (Dekel et al., 2012) and so on. In reality, it could be difficult to collect fully supervised data, hence a number of methods have been proposed to deal with various online weakly supervised learning problems, such as online complementary-label learning (Kaneko et al., 2019), online positive-unlabeled learning (Zhang et al., 2020), online learning with noisy data (Natarajan et al., 2013), and online partial-label learning (Wang et al., 2020). However, it is hard to apply these methods to our online SD classification, because they can not deal with the given data pairs. Therefore, we provide the first attempt to investigate online SD classification in this paper.

3 The proposed methods

In this section, we investigate effective solutions for online SD classification. We first present an intuitive adaptation of the unbiased risk estimator to the online learning scenario with a conservative regularization term (Li et al., 2014). However, this intuitive method suffers from the problem that the objective function could go to negative infinity (i.e., unbounded from below). To address this problem, we further propose two corresponding improvements where one employs a linearly scaled cost and the other one employ a quadratically scaled cost. Therefore, the two improved methods are expected to achieve better performance than the original naive method.

3.1 Online gradient descent method

Following Shimada et al. (2020), we adopt the same generation process of similar and dissimilar data. Let $\{(\boldsymbol{x}_t, \boldsymbol{x}'_t, s_t)\}_{t=1}^{T}$ be a sequence of examples (arriving in sequential order) sampled from the data distribution of SD classification, where $(\boldsymbol{x}_t, \boldsymbol{x}'_t)$ and $s_t \in \{-1, +1\}$ denote the data pair and the similarity label (i.e., the data pair is similar if $s_t = +1$ and is dissimilar if $s_t = -1$) received at the t-th round, and T is the total number of rounds during the training phase. In this paper, we aim to incrementally update a linear classifier $f(\boldsymbol{x}_t) = \boldsymbol{w}^\top \boldsymbol{x}_t$ for online SD classification.

Motivated by the unbiased risk estimator of SD classification in Eq. (1), we propose to employ the following risk for the example $(\boldsymbol{x}_t, \boldsymbol{x}'_t, s_t)$ received at the t-th round:

$$R_t^{\mathrm{SD}} = \frac{1+s_t}{2} R_t^{\mathrm{S}} + \frac{1-s_t}{2} R_t^{\mathrm{D}}, \tag{2}$$

where $R_t^{\mathrm{S}} = \pi_{\mathrm{S}}\big(L(f(\boldsymbol{x}_t)) + L(f(\boldsymbol{x}'_t))\big)$ and $R_t^{\mathrm{D}} = \pi_{\mathrm{D}}\big(L(-f(\boldsymbol{x}_t)) + L(-f(\boldsymbol{x}'_t))\big)$. It is noteworthy that Eq. (2) may not be convex even if a convex loss function $\ell(\cdot)$ (e.g., the hinge loss) is used. Fortunately, if the used loss function $\ell(\cdot)$ has the linear-odd property, i.e., $\ell(z) - \ell(-z) = -z$, we can easily verify that Eq. (2) is convex. As shown by previous studies (Bao et al., 2018; Shimada et al., 2020), among all the losses satisfying the linear-odd property, the *double hinge loss* (i.e., $\ell(z) = \max(-z, \max(0, 1/2 - z/2))$) achieves highly competitively performance. Hence we use double hinge loss as the surrogate loss function in this paper. By substituting $f(\cdot)$ and $L(\cdot)$ with their expressions into R_t^{S} and R_t^{D}, we obtain

$$R_t^{\mathrm{S}}(\boldsymbol{w}) = \frac{\pi_{\mathrm{S}}}{\pi_+ - \pi_-}\big[-\pi_-(\boldsymbol{w}^\top \boldsymbol{x}_t + \boldsymbol{w}^\top \boldsymbol{x}'_t) + (\pi_+ - \pi_-)\big(\ell(\boldsymbol{w}^\top \boldsymbol{x}_t) + \ell(\boldsymbol{w}^\top \boldsymbol{x}'_t)\big)\big], \tag{3}$$

$$R_t^{\mathrm{D}}(\boldsymbol{w}) = \frac{\pi_{\mathrm{D}}}{\pi_+ - \pi_-}\big[\pi_-(\boldsymbol{w}^\top \boldsymbol{x}_t + \boldsymbol{w}^\top \boldsymbol{x}'_t) + (\pi_+ - \pi_-)\big(\ell(-\boldsymbol{w}^\top \boldsymbol{x}_t) + \ell(-\boldsymbol{w}^\top \boldsymbol{x}'_t)\big)\big]. \tag{4}$$

By further leveraging the widely used conservative regularization in online learning (Li et al., 2014; Zhang et al., 2020), we derive the following online update strategy:

$$\boldsymbol{w}_{t+1} = \arg\min_{\boldsymbol{w} \in \mathcal{W}} R_t^{\mathrm{SD}}(\boldsymbol{w}) + \frac{1}{2\gamma}\|\boldsymbol{w} - \boldsymbol{w}_t\|_2^2, \tag{5}$$

where $R_t^{\mathrm{SD}}(\boldsymbol{w}) = \frac{1+s_t}{2} R_t^{\mathrm{S}}(\boldsymbol{w}) + \frac{1-s_t}{2} R_t^{\mathrm{D}}(\boldsymbol{w})$, γ is a positive penalty parameter, and $\mathcal{W}$ is the convex feasible set of $\boldsymbol{w}$ (which means $\boldsymbol{w}$ should be always updated within this set). Let

$\mathcal{L}_t(\boldsymbol{w})$ represent the right hand side of the equality in Eq. (5), then we can derive its gradient as follows:

$$\nabla \mathcal{L}_t(\boldsymbol{w}) = \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) + \frac{1}{\gamma}(\boldsymbol{w} - \boldsymbol{w}_t),$$

where $\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = \frac{1+s_t}{2}\nabla R_t^{\mathrm{S}}(\boldsymbol{w}) + \frac{1-s_t}{2}\nabla R_t^{\mathrm{D}}(\boldsymbol{w})$ and

$$\nabla R_t^{\mathrm{S}}(\boldsymbol{w}) = \pi_{\mathrm{S}}\big(\ell'(\boldsymbol{w}^\top \boldsymbol{x}_t) + \ell'(\boldsymbol{w}^\top \boldsymbol{x}'_t)\big) - \frac{\pi_{\mathrm{S}}\pi_-}{\pi_+ - \pi_-}(\boldsymbol{x}_t + \boldsymbol{x}'_t),$$

$$\nabla R_t^{\mathrm{D}}(\boldsymbol{w}) = \pi_{\mathrm{D}}\big(\ell'(-\boldsymbol{w}^\top \boldsymbol{x}_t) + \ell'(-\boldsymbol{w}^\top \boldsymbol{x}'_t)\big)\frac{\pi_{\mathrm{D}}\pi_-}{\pi_+ - \pi_-}(\boldsymbol{x}_t + \boldsymbol{x}'_t).$$

Here, we list the specific values of $\ell'(\boldsymbol{w}^\top \boldsymbol{x}_t)$ and $\ell'(-\boldsymbol{w}^\top \boldsymbol{x}_t)$ in Table 1. By setting $\nabla \mathcal{L}_t(\boldsymbol{w})$ to zero, the update rule can be expressed as

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \gamma \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}).$$

It is not surprising to verify that the above update rule is actually equivalent to the following update rule:

$$\boldsymbol{w}_{t+1} = \arg\min_{\boldsymbol{w}\in\mathcal{W}} \sum_{i=1}^{t} R_i^{\mathrm{SD}}(\boldsymbol{w}) + \frac{1}{2\gamma}\|\boldsymbol{w}\|_2^2,$$

which is exactly the *follow-the-regularized-leader* procedure with Euclidean regularization (also known as the *online gradient descent* (OGD) algorithm) (Shalev-Shwartz, 2011). We therefore name this method OSD-OGD for short.

Regret bound for OSD-OGD Here, we analyze the *regret bound* of our proposed OSD-OGD method.

Theorem 1 *Let $\mathcal{L}_1, \ldots, \mathcal{L}_T$ be a sequence of convex functions such that $\mathcal{L}_t$ is ρ_t-Lipschitz with respect to $\|\cdot\|_2$. Let ρ satisfy the condition that $\frac{1}{T}\sum_{t=1}^{T}\rho_t^2 \le \rho^2$. Let $\boldsymbol{w}_\star = \arg\min_{\boldsymbol{w}\in\mathcal{W}}\sum_{t=1}^{T}\mathcal{L}_t(\boldsymbol{w})$ be optimal hypothesis derived by batch learning (empirical risk minimization). Suppose for every hypothesis $\boldsymbol{w}\in\mathcal{W}$, it satisfies $\|\boldsymbol{w}\|_2 \le B$ and $\gamma = \frac{B}{\rho\sqrt{2T}}$. Then, we derive the following regret bound:*

$$\sum_{t=1}^{T}\mathcal{L}_t(\boldsymbol{w}_t) - \sum_{t=1}^{T}\mathcal{L}_t(\boldsymbol{w}_\star) \le B\rho\sqrt{2T}.$$

The proof is provided in Appendix 1. From Theorem 1, we show that the regret bound of OSD-OGD grows sublinearly with T. Specifically, after a certain number of iterations, the models obtained by our method could be close to the ideal solution $\boldsymbol{w}_\star$.

Table 1 The specific values of $\ell'(\boldsymbol{w}^\top \boldsymbol{x})$ and $\ell'(-\boldsymbol{w}^\top \boldsymbol{x})$

	$\boldsymbol{w}^\top \boldsymbol{x}_t \ge 1$	$-1 < \boldsymbol{w}^\top \boldsymbol{x}_t < 1$	$\boldsymbol{w}^\top \boldsymbol{x}_t \le -1$
$\ell'(\boldsymbol{w}^\top \boldsymbol{x}_t)$	0	$-\frac{1}{2}\boldsymbol{x}_t$	$-\boldsymbol{x}_t$
$\ell'(-\boldsymbol{w}^\top \boldsymbol{x}_t)$	$\boldsymbol{x}_t$	$\frac{1}{2}\boldsymbol{x}_t$	0

3.2 Linearly scaled passive-aggressive method

There is an important issue in the above OSD-OGD algorithm. In Eq. (5), the value of the whole objective function could go to negative infinity (i.e., unbounded from below) due to a negative term in $R_t^{\mathrm{S}}(\boldsymbol{w})$, which means we may still focus too much on the example whose cost $R_t^{\mathrm{SD}}(\boldsymbol{w})$ is very small (even less than zero). As shown in previous works (Kiryo et al., 2017; Lu et al., 2019), this problem would seriously degrade classification accuracy. Inspired by the margin criterion that only focuses on examples with small margin, we aim to focus more on examples with larger cost and no longer update the classifier parameter when receiving examples whose costs become negative. In this way, we further propose an improved online SD classification method that can well address the above issue. Specifically, at the t-th round, we propose the following update method:

$$\boldsymbol{w}_{t+1} = \arg\min_{\boldsymbol{w}\in\mathcal{W}} \frac{1}{2}\|\boldsymbol{w}-\boldsymbol{w}_t\|_2^2 + C\xi, \qquad \text{s.t.} \quad R_t^{\mathrm{SD}}(\boldsymbol{w}) \le \xi, \quad \xi \ge 0, \tag{6}$$

where C is a positive parameter that controls the influence of the slack term (i.e., ξ) on the objective function. In other words, a larger value of C implies a more *aggressive* update step (Crammer et al., 2006), and the classifier would not be updated (i.e., *passive*) if the cost $R_t^{\mathrm{SD}}(\boldsymbol{w})$ is smaller than zero. Note that the cost $R_t^{\mathrm{SD}}(\boldsymbol{w})$ scales *linearly* with the slack variable ξ. Therefore, the minimization of Eq. (6) results in a *linearly scaled passive-aggressive* algorithm for online SD classification (OSD-LSPA for short). Interestingly, it is also worth noting that the constraint in Eq. (6) plays the same role as the non-negative risk estimator (Kiryo et al., 2017) in the training process. That is, Eq. (6) can be also equivalently expressed as $\boldsymbol{w}_{t+1} = \arg\min_{\boldsymbol{w}\in\mathcal{W}} \frac{1}{2}\|\boldsymbol{w}-\boldsymbol{w}_t\|_2^2 + C\max\{R_t^{\mathrm{SD}}(\boldsymbol{w}), 0\}$. To solve the problem in Eq. (6), we define the Lagrangian of Eq. (6) as follows:

$$\mathcal{L}_t(\boldsymbol{w},\tau,\xi,\eta) = \frac{1}{2}\|\boldsymbol{w}-\boldsymbol{w}_t\|_2^2 + C\xi + \tau(R_t^{\mathrm{SD}}(\boldsymbol{w}) - \xi) - \eta\xi. \tag{7}$$

Setting the partial derivatives of $\mathcal{L}_t(\boldsymbol{w},\tau,\xi,\eta)$ with respect to $\boldsymbol{w}$ and ξ to zero, we can easily obtain $\boldsymbol{w} = \boldsymbol{w}_t - \tau\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})$ and $\tau + \eta = C$. Then, we introduce the following equivalent expression of $R_t^{\mathrm{SD}}(\boldsymbol{w})$:

$$R_t^{\mathrm{SD}}(\boldsymbol{w}) = \boldsymbol{w}^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) + A,$$

where A is a constant. The above equality holds because $R_t^{\mathrm{SD}}(\boldsymbol{w})$ is a first-order function with respect to $\boldsymbol{w}$. Furthermore, we can derive the specific values of A so that the above equality exactly holds under various conditions. We specially list all the specified values of A under various conditions in Table 2. As can be seen from Table 2, we can easily verify that under all the possible conditions, and we can always find a suitable value of A that makes the equality $R_t^{\mathrm{SD}}(\boldsymbol{w}) = \boldsymbol{w}^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) + A$ true. By solving the problem of Eq. (7) with respect to τ, we can obtain

$$\tau = \frac{A + \boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w})}{\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2}. \tag{8}$$

It is worth noting that the KKT conditions confine η and τ to be non-negative, hence we conclude that τ and η should satisfy $0 \le \tau \le C$ and $0 \le \eta \le C$ respectively. By taking into

Table 2 List of specific values of A that make $R_t^{\mathrm{SD}}(\boldsymbol{w}) = A + \boldsymbol{w}^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w})$

$s_t \boldsymbol{w}^\top \boldsymbol{x}_t$	$s_t \boldsymbol{w}^\top \boldsymbol{x}_t'$	A	$R_t^{\mathrm{SD}}(\boldsymbol{w})$ & $\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})$
≥ 1	≥ 1	0	$R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} \boldsymbol{w}^\top(\pi_- \boldsymbol{x}_t + \pi_- \boldsymbol{x}_t')$
			$\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} (\pi_- \boldsymbol{x}_t + \pi_- \boldsymbol{x}_t')$
≥ 1	$-1 < \cdot < 1$	$\frac{(1+s_t)\pi_{\mathrm{S}}+(1-s_t)\pi_{\mathrm{D}}}{4}$	$R_t^{\mathrm{SD}}(\boldsymbol{w}) = A - \frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} \boldsymbol{w}^\top(\pi_- \boldsymbol{x}_t + \frac{1}{2}\boldsymbol{x}_t')$
			$\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} (\pi_- \boldsymbol{x}_t + \frac{1}{2}\boldsymbol{x}_t')$
≥ 1	≤ -1	0	$R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} \boldsymbol{w}^\top(\pi_- \boldsymbol{x}_t + \pi_+ \boldsymbol{x}_t')$
			$\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} (\pi_- \boldsymbol{x}_t + \pi_+ \boldsymbol{x}_t')$
$-1 < \cdot < 1$	≥ 1	$\frac{(1+s_t)\pi_{\mathrm{S}}+(1-s_t)\pi_{\mathrm{D}}}{4}$	$R_t^{\mathrm{SD}}(\boldsymbol{w}) = A - \frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} \boldsymbol{w}^\top(\frac{1}{2}\boldsymbol{x}_t + \pi_- \boldsymbol{x}_t')$
			$\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} (\frac{1}{2}\boldsymbol{x}_t + \pi_- \boldsymbol{x}_t')$
$-1 < \cdot < 1$	$-1 < \cdot < 1$	$\frac{(1+s_t)\pi_{\mathrm{S}}+(1-s_t)\pi_{\mathrm{D}}}{2}$	$R_t^{\mathrm{SD}}(\boldsymbol{w}) = A - \frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} \boldsymbol{w}^\top(\frac{1}{2}\boldsymbol{x}_t + \frac{1}{2}\boldsymbol{x}_t')$
			$\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} (\frac{1}{2}\boldsymbol{x}_t + \frac{1}{2}\boldsymbol{x}_t')$
$-1 < \cdot < 1$	≤ -1	$\frac{(1+s_t)\pi_{\mathrm{S}}+(1-s_t)\pi_{\mathrm{D}}}{4}$	$R_t^{\mathrm{SD}}(\boldsymbol{w}) = A - \frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} \boldsymbol{w}^\top(\frac{1}{2}\boldsymbol{x}_t + \pi_+ \boldsymbol{x}_t')$
			$\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} (\frac{1}{2}\boldsymbol{x}_t + \pi_+ \boldsymbol{x}_t')$
≤ -1	≥ 1	0	$R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} \boldsymbol{w}^\top(\pi_+ \boldsymbol{x}_t + \pi_- \boldsymbol{x}_t')$
			$\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} (\pi_+ \boldsymbol{x}_t + \pi_- \boldsymbol{x}_t')$
≤ -1	$-1 < \cdot < 1$	$\frac{(1+s_t)\pi_{\mathrm{S}}+(1-s_t)\pi_{\mathrm{D}}}{4}$	$R_t^{\mathrm{SD}}(\boldsymbol{w}) = A - \frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} \boldsymbol{w}^\top(\pi_+ \boldsymbol{x}_t + \frac{1}{2}\boldsymbol{x}_t')$
			$\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} (\pi_+ \boldsymbol{x}_t + \frac{1}{2}\boldsymbol{x}_t')$
≤ -1	≤ -1	0	$R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} \boldsymbol{w}^\top(\pi_+ \boldsymbol{x}_t + \pi_+ \boldsymbol{x}_t')$
			$\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) = -\frac{(1+s_t)\pi_{\mathrm{S}}-(1-s_t)\pi_{\mathrm{D}}}{2(\pi_+-\pi_-)} (\pi_+ \boldsymbol{x}_t + \pi_+ \boldsymbol{x}_t')$

account Eq. (8) and denoting the finally specified value of τ at the t-th round as τ_t, we have the three cases:

$$\begin{cases} \tau_t = 0, & \text{if } \frac{A+\boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w})}{\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2} < 0 \\ \tau_t = \frac{A+\boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w})}{\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2}, & \text{if } 0 \leq \frac{A+\boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w})}{\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2} \leq C \\ \tau_t = C, & \text{if } \frac{A+\boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w})}{\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2} > C \end{cases} \tag{9}$$

For the above three cases, the first two cases can be easily verified. Therefore, we only explain in detail how the third case comes. When $\frac{A+\boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w})}{\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2} > C$, we obtain

$$C\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2 < A + \boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}). \tag{10}$$

Besides, we also know that the constraint in Eq. (6) must hold at the optimum, and thus $R_t^{\mathrm{SD}}(\boldsymbol{w}) \leq \xi$. By further considering that $\boldsymbol{w} = \boldsymbol{w}_t - \tau \nabla R_t^{\mathrm{SD}}(\boldsymbol{w})$ and $R_t^{\mathrm{SD}}(\boldsymbol{w}) = \boldsymbol{w}^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) + A$, we obtain

$$A + \boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) - \tau\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2 \leq \xi. \tag{11}$$

By combining Eqs. (10) and (11), we have $(C-\tau)\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2 < \xi$. In this way, we can conclude that $\xi > 0$, due to $0 \le \tau \le C$. According to the KKT complementarity condition, we know that $\eta\xi = 0$ at the optimum. Since ξ is strictly positive, we get that η must equal zero. Recall that $\tau + \eta = C$, which further ensures that $\tau = C$. Therefore, the derivation process of the third case is completed.

Integrating the three possible cases in Eq. (9) into a single equation, we obtain

$$\tau_t = \min\left(C, \max\left(0, \frac{A + \boldsymbol{w}_t^{\top}\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})}{\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2}\right)\right),$$

where the detailed information of A and $\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})$ is provided in Table 2. In summary, the update method of our OSD-LSPA algorithm is presented as follows:

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \tau_t \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}).$$

Mistake bound of OSD-LSPA Here, we analyze the *mistake* bound of our proposed OSD-LSPA algorithm.

Theorem 2 *Let $E_t(\boldsymbol{w}_t)$ denote whether the prediction with $\boldsymbol{w}_t$ on $\boldsymbol{x}_t$ is a mistake (i.e., $E_t(\boldsymbol{w}_t) = \mathbb{I}[\mathrm{sign}(\boldsymbol{w}_t^{\top}\boldsymbol{x}_t) \neq y_t]$). Suppose $\left\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\right\|_2 \le r$ for all t and $R_t^{\mathrm{SD}}(\boldsymbol{w}_t) \ge G$ $(G \ge 0)$ if $E_t(\boldsymbol{w}_t) = 1$. Then, for any vector $\boldsymbol{v} \in \mathcal{W}$, the number of prediction mistakes made by OSD-LSPA is upper-bounded by*

$$\sum_{t=1}^{T} E_t(\boldsymbol{w}_t) \le \max\left(\frac{1}{CG}, \frac{r^2}{G^2}\right)\left(\|\boldsymbol{v}\|_2^2 + 2C\sum_{t=1}^{T} R_t^{\mathrm{SD}}(\boldsymbol{v})\right).$$

The proof is provided in Appendix 2. Theorem 2 provides a direct bound on the number of mistakes made by OSD-LSPA. From Theorem 2, we can observe that when G becomes larger, OSD-LSPA would make fewer mistakes.

3.3 Quadratically scaled passive-aggressive method

We also propose another improvement that serves as an alternative for Eq. (6), which makes the cost $R_t^{\mathrm{SD}}(\boldsymbol{w})$ scale *quadratically* with ξ, which results in a new objective function as follows:

$$\boldsymbol{w}_{t+1} = \arg\min_{\boldsymbol{w}\in\mathcal{W}} \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{w}_t\|_2^2 + C\xi^2, \qquad \text{s.t.} \quad R_t^{\mathrm{SD}}(\boldsymbol{w}) \le \xi. \tag{12}$$

Note that the cost $R_t^{\mathrm{SD}}(\boldsymbol{w})$ scales *quadratically* with the slack variable ξ. Therefore, the minimization of Eq. (12) results in a *quadratically scaled passive-aggressive* algorithm for online SD classification (OSD-QSPA for short). It is noteworthy that the constraint $\xi \ge 0$ is no longer required since ξ^2 is always non-negative. To solve the problem in Eq. (12), we define the Lagrangian of Eq. (12) as follows:

$$\mathcal{L}_t(\boldsymbol{w}, \tau, \xi) = \frac{1}{2}\|\boldsymbol{w} - \boldsymbol{w}_t\|_2^2 + C\xi^2 + \tau(R_t^{\mathrm{SD}}(\boldsymbol{w}) - \xi). \tag{13}$$

By solving the problem of Eq. (13) with respect to τ, we obtain

$$\tau = \frac{2C(A + \boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}))}{2C\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2 + 1},$$

where $R_t^{\mathrm{SD}}(\boldsymbol{w}) = \boldsymbol{w}^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) + A$ is used, and the possible values of A are provided in Table 2. Note that the KKT conditions confine τ to be non-negative. Therefore, we obtain the following specified value of τ at the t-th round:

$$\tau_t = \max\left(0, \frac{2C(A + \boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}))}{2C\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2 + 1}\right).$$

In this way, the update method of OSD-QSPA is given as

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \tau_t \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}).$$

Relative loss bound for OSD-QSPA We provide a relative loss bound for the OSD-QSPA method.

Theorem 3 *Let* $\{(\boldsymbol{x}_t, \boldsymbol{x}'_t, s_t)\}_{t=1}^T$ *be the received training pairs. Assume that* $\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\|^2 \le r^2$ *for all* t. *Then for any vector* $\boldsymbol{v} \in \mathcal{W}$, *the cumulative squared loss of OSD-QSPA on this sequence of examples is upper-bounded by*

$$\sum_{t=1}^T \left(R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\right)^2 \le \left(r^2 + \frac{1}{2C}\right)\left(\|\boldsymbol{v}\|_2^2 + 2C\sum_{t=1}^T (R_t^{\mathrm{SD}}(\boldsymbol{v}))^2\right).$$

The proof is provided in Appendix 3. As shown in previous works (Crammer et al., 2006; Kaneko et al., 2019), the cumulative squared loss serves as an upper bound of the number of prediction mistakes.

3.4 Class prior estimation

As the fraction of similar data pairs π_{S} and class prior π_+ may not always be available in online learning. Thus, we provide the following steps to estimate π_{S} and π_+.

Estimation of π_S To empirically estimate π_{S}, it is straightforward for us to consider an iteratively *unbiased* updated process. Specifically, let us denote $\hat{\pi}_{\mathrm{S},t}$ as the estimation of π_{S} at the t-th round. Then, we can obtain the following update rule:

$$\hat{\pi}_{\mathrm{S},t} = \frac{(t-1)\hat{\pi}_{\mathrm{S},t-1} + \frac{s_t+1}{2}}{t},$$

where $\hat{\pi}_{\mathrm{S},0}$ is set to 0. However, according the equality $\pi_{\mathrm{S}} = \pi_+^2 + \pi_-^2$, we find that $\pi_{\mathrm{S}} = 2(\pi_+ - \frac{1}{2})^2 + \frac{1}{2} \ge 0.5$ holds for any π_+. Note that $\pi_{\mathrm{S}} = 0.5$ if and only if $\pi_+ = 0.5$, which is impossible to calculate the risk according Eqs. (3) and (4), since $\pi_+ - \pi_- = 0$. Therefore, we further propose to correct $\hat{\pi}_{\mathrm{S},t}$ by the following rule:

$$\widetilde{\pi}_{\mathrm{S},t} = \begin{cases} \hat{\pi}_{\mathrm{S},t}, & \hat{\pi}_{\mathrm{S},t} > 0.5, \\ 0.5 + \epsilon, & \hat{\pi}_{\mathrm{S},t} \le 0.5, \end{cases}$$

where ϵ is set to 10^{-3} to ensure that $\widetilde{\pi}_{\mathrm{S},t} > 0.5$.

Estimation of π_+. To empirically estimate π_+, we assume that the positive class prior should be larger than the negative class prior, i.e., $\pi_+ > 0.5$. Note that if $\pi_+ < 0.5$, we can

switch the role of positive class and negative class. Based on the equality $\pi_S = \pi_+^2 + \pi_-^2$, the class prior π_+ at the t-th round can be estimated as

$$\widetilde{\pi}_{+,t} = \frac{1 + \sqrt{2\widetilde{\pi}_{S,t} - 1}}{2}.$$

4 Experiments

In this section, we conduct extensive experiments to demonstrate the effectiveness of our proposed methods OSD-OGD, OSD-LSPA, and OSD-QSPA on various benchmark datasets (regular-scale and large-scale). Then, we provide the hyper-parameter analysis and the influence of the round T on the performance of our proposed methods.

4.1 Experimental setup

Datasets We use nine benchmark datasets, including ionosphere, magic, phishing, spambase, steel-plates, wdbc, usps, pendigits, and cnae-9. These datasets can be downloaded from the UCI Machine Learning Repository (Blake and Merz, 1998). We also use two large-scale datasets, MNIST (LeCun et al., 1998) and Fashion-MNIST (Xiao et al., 2017). The brief statistics of these datasets are provided in Table 3. Since usps, pendigits, cnae-9, MNIST, and Fashion-MNIST are used for multi-class classification, we manually transformed them into binary classification datasets. Specifically, for usps and pendigits, we regard the even digits as the positive class and the odd digits as the negative class. For cnae-9, the positive class is formed by "2", "4", "6", and "8", the negative class is formed by "1", "3", "5", "7", and "9". For MNIST, we divide the multi-class dataset into nine binary datasets, including MNIST-0vs1, MNIST-0vs2, …, and MNIST-0vs9, which the "0" digit is used for the positive class, and another digit is used for the negative class (i.e., one dataset for each digit "1"…"9"). For Fashion-MNIST, the positive class is formed by "T-shirt", "pullover", "coat", "shirt", and "bag", the negative class is formed by "trouser", "dress", "sandal", "sneaker", and "ankle boot".

Table 3 Brief statistics of UCI datasets for binary classification

Dataset	# Examples	# Features	# Classes
ionosphere	351	33	2
magic	19,020	10	2
phishing	11,055	30	2
spambase	4601	57	2
steel-plates	1941	33	2
wdbc	569	30	2
usps	9298	256	10
pendigits	10,992	16	10
cnae-9	1080	856	9
MNIST	70,000	784	10
Fashion	70,000	784	10

Preprocessing For each dataset, we randomly sample 80% examples from the whole dataset as training examples, and the rest 20% examples are taken as test examples that are used to test the performance of the trained model. To convert ordinarily labeled training data into similar and dissimilar data pairs, we first set the positive prior π_+ to 0.7 in our experiments, so that we obtain the fraction of similar (dissimilar) data pairs $\pi_S = \pi_+^2 + \pi_-^2 = 0.58$ ($\pi_D = 1 - 2\pi_+\pi_- = 0.42$). Then, we randomly sample pairwise similar and dissimilar data pairs following the ratios of π_S and π_D. The training set for SD classification is constructed when no similar or dissimilar data pairs can be generated from the ordinary training set. It is worth noting that the ratios of π_S and the class prior p_+ are unknown in practice. Fortunately, we could estimate them by using the method provided in Sect. 3.4. To precisely evaluate the performance, we report the mean classification accuracy with standard deviation over 10 trials.

Methods We proposed the following methods for online SD classification: **OSD-OGD** (ours): A simple online algorithm combines the unbiased risk estimator of SD classification and conservative regularization, and results in an online gradient descent algorithm. The penalty parameter γ is chosen from $\{10^{-1}, 10^{-2}, \ldots, 10^{-6}\}$ on UCI datasets, and $\{10^{-5}, 10^{-6}, \ldots, 10^{-10}\}$ on large-scale datasets. **OSD-LSPA** (ours): An online passive-aggressive algorithm with linearly scaled cost. The aggressiveness parameter C is chosen from $\{10^{-1}, 10^{-2}, \ldots, 10^{-6}\}$ on UCI datasets, and $\{10^{-5}, 10^{-6}, \ldots, 10^{-10}\}$ on large-scale datasets. **OSD-QSPA** (ours): An online passive-aggressive algorithm with quadratically scaled cost. The aggressiveness parameter C is chosen from $\{10^{-1}, 10^{-2}, \ldots, 10^{-6}\}$ on UCI datasets, and $\{10^{-5}, 10^{-6}, \ldots, 10^{-10}\}$ on large-scale datasets. We compare our proposed methods with the following methods: **A-Ramp** (Jian et al., 2018): The online learning with label noise by using an adaptive ramp loss ($\ell(z) = \min\{1 - s, \max\{0, 1 - z\}\}$). The setting of noise-resilient parameter s follows the original paper. Here, we use the binary version of **A-Ramp** by treating similar data as positive data and dissimilar data as negative data. **OBC (supervised)** (Shalev-Shwartz, 2011): The *online binary classification* with fully labeled data. The regularization parameter of OBC (supervised) is also chosen from $\{10^{-1}, 10^{-2}, \ldots, 10^{-6}\}$ on UCI datasets, and $\{10^{-5}, 10^{-6}, \ldots, 10^{-10}\}$ on large-scale datasets. **KM** (MacQueen et al., 1967): The *k-means* clustering method with $k = 2$. For KM, the pairwise information is ignored. **CKM** (Wagstaff et al., 2001): The *constrained k-means clustering* method with $k = 2$. CKM uses pairwise (dissimilar) information as must-link (cannot-link) constraints. For the two clustering methods, the classification accuracy is evaluated by $1 - \min(r, 1 - r)$, where r denotes the error rate. For a fair comparison, our proposed methods and OBC are trained with the double hinge loss (i.e., $\ell(z) = \max(-z, \max(0, 1/2 - z/2))$). Hence OBC can be regarded as the performance ceiling of our proposed methods.

4.2 Experimental results

Performance on UCI datasets Tables 4 and 5 report the classification accuracy of different methods on UCI datasets, using 100% and 25% of the training set to generate similar and dissimilar data pairs respectively. The best performance is highlighted in bold and the second best performance is underlined. From the two tables, we have the following observations: Our proposed methods clearly outperform the compared baselines, which implies that our proposed methods are effective in handling online SD classification. Among our proposed methods, OSD-LSPA has the same level of (maybe slightly better) performance compared with OSD-QSPA, as they both are variants of the passive-aggressive algorithm.

Table 4 Classification accuracy (mean±std) of each method on UCI datasets by using 100% of the training set to generate similar and dissimilar data pairs

Dataset	OSD-LSPA	OSD-QSPA	OSD-OGD	A-Ramp	KM	CKM	OBC (supervised)
ionosphere	**0.816±0.068**	0.803 ± 0.056	0.750±0.063	0.685±0.000	0.720±0.051	0.725±0.035	0.881±0.024
magic	**0.789±0.007**	0.783 ± 0.005	0.783 ± 0.007	0.762±0.011	0.548±0.007	0.527±0.010	0.790±0.004
phishing	**0.916±0.006**	**0.916±0.008**	0.915±0.011	0.810±0.012	0.645±0.011	0.597±0.056	0.927±0.004
spambase	**0.897±0.009**	0.890±0.012	0.895 ± 0.013	0.529±0.014	0.692±0.004	0.691±0.102	0.921±0.011
steel-plates	**0.973±0.033**	0.870±0.042	0.895 ± 0.048	0.783±0.003	0.520±0.007	0.515±0.017	0.999±0.001
wdbc	0.940±0.018	0.941 ± 0.021	0.932±0.039	**0.948±0.004**	0.921±0.026	0.907±0.026	0.975±0.013
usps	0.859 ± 0.084	**0.861±0.049**	0.799±0.064	0.700±0.000	0.548±0.010	0.601±0.014	0.937±0.014
pendigits	**0.837±0.032**	0.829 ± 0.050	0.788±0.037	0.701± 0.002	0.623±0.035	0.682±0.024	0.871±0.019
cnae9	0.807 ± 0.037	**0.814±0.043**	0.798±0.030	0.701±0.002	0.592±0.073	0.536±0.030	0.912±0.028

Table 5 Classification accuracy (mean±std) of each method on UCI datasets by using 25% of the training set to generate similar and dissimilar data pairs

Dataset	OSD-LSPA	OSD-QSPA	OSD-OGD	A-Ramp	KM	CKM	OBC (supervised)
ionosphere	**0.764±0.144**	0.733 ± 0.079	0.695±0.098	0.656±0.001	0.694±0.056	0.664±0.055	0.816±0.068
magic	**0.783±0.007**	0.769±0.011	0.776 ± 0.004	0.758±0.000	0.547±0.008	0.520±0.011	0.785±0.005
phishing	**0.910±0.01**1	0.894±0.012	0.898 ± 0.011	0.798±0.008	0.635±0.024	0.564±0.037	0.914±0.004
spambase	**0.887±0.018**	0.869±0.017	0.873 ± 0.018	0.838±0.003	0.692±0.007	0.639±0.106	0.906±0.009
steel-plates	**0.780±0.053**	0.768±0.048	0.745±0.066	0.411±0.002	0.522±0.012	0.510±0.024	0.977±0.032
wdbc	30.930 ± 0.053	**0.934±0.043**	0.915±0.047	0.750±0.053	0.910±0.031	0.897±0.089	0.963±0.019
usps	0.795 ± 0.073	**0.815±0.052**	0.765±0.060	0.700±0.000	0.545±0.008	0.592±0.020	0.922±0.026
pendigits	0.746±0.055	**0.799±0.038**	0.756 ± 0.046	0.702±0.008	0.622±0.046	0.620±0.068	0.867±0.011
cnae9	0.724 ± 0.036	**0.745±0.023**	0.611±0.160	0.705±0.008	0.668±0.058	0.557±0.087	0.850±0.034

OSD-LSPA and OSD-QSPA outperform OSD-OGD in most cases, which supports our motivation that OSD-LSPA and OSD-QSPA can address the problem of negative objective function in OSD-OGD. Moreover, the performance of the noise-resilient method A-Ramp is worse than our proposed methods. This observation indicates that even noise-resilient online learning methods could not deal with the online SD classification problem, but our proposed methods could effectively solve this problem. By comparing the performance of our methods in Tables 4 and 5, we can find that the performance will be better if more training data pairs (training rounds) are provided, which clearly supports our derived regret, mistake, and relative loss bounds (in Theorems 1–3) that the bounds become tighter if the number of training rounds T increases.

Performance on large-scale datasets Tables 6 and 7 report the classification accuracy of different methods on large-scale datasets, using 100% and 25% of the training set to generate similar and dissimilar data pairs respectively. The best performance is highlighted in bold and the second best performance is underlined. From the two tables, we have the following observations: OSD-OGD performs worse on large-scale datasets, which implies that the negative objective in OSD-OGD causes a more serious negative impact on large-scale datasets. OSD-LSPA and OSD-QSPA significantly outperform OSD-OGD and OSD-LSPA performance the best in all cases, which can support that our proposed passive-aggressive methods are effective in handling large-scale data. Interestingly, A-Ramp seems not able to learn anything in all cases on large-scale datasets. This observation also indicates that even noise-resilient online learning methods could not handle online SD classification on large-scale datasets. As shown in Tables 6 and 7, the performance of our proposed methods will also be better if more training data pairs are provided, which supports Theorems 1–3.

4.3 Further analysis

Parameter sensitivity We further conduct parameter sensitivity analysis of C (used in OSD-LSPA and OSD-QSPA) and γ (used in OSD-OGD) to show the effect of the parameter on the proposed methods. As can be seen from Fig. 1, the parameters C and γ have a great influence on the performance. Firstly, exceedingly large or small parameters C and γ can significantly reduce the effectiveness of the proposed methods. Secondly, parameters C and γ for large-scale datasets (such as MNIST and Fashion-MNIST) would be smaller than the UCI datasets since even a small parameter can effectively update the model on large-scale datasets due to the high dimensions. These observations validate that a suitable learning rate is the key to gradient update.

Increasing T As shown by Theorems 1–3, the performance of our online SD classification methods is expected to be improved if the number of training rounds T increases. To empirically validate such theoretical findings, we further conduct experiments on four large-scale datasets (i.e., MNIST-0vs3, MNIST-0vs5, MNIST-0vs7, and MNIST-0vs9) by increasing T. As shown in Fig. 2, the classification accuracy of OSD-LSPA and OSD-QSPA increases steadily and tends to converge with the increase of T. The classification accuracy of OSD-OGD also increases when T is small. However, the performance of OSD-OGD becomes unstable when T becomes larger, due to the influence of negative objective function in OSD-OGD. These observations are clearly in accordance with our provided Theorems 1–3.

Table 6 Classification accuracy (mean±std) of each method on large-scale datasets by using 100% of the training set to generate similar and dissimilar data pairs

Dataset	OSD-LSPA	OSD-QSPA	OSD-OGD	A-Ramp	KM	CKM	OBC (supervised)
MNIST-0vs1	**0.998±0.001**	**0.998±0.001**	0.935±0.018	0.700±0.000	0.989±0.001	0.946±0.002	1.000±0.000
MNIST-0vs2	**0.984±0.008**	0.965 ± 0.009	0.875±0.026	0.707±0.013	0.895±0.004	0.841±0.005	0.990±0.003
MNIST-0vs3	**0.989±0.007**	0.988 ± 0.004	0.885±0.034	0.701±0.006	0.872±0.003	0.838±0.004	0.996±0.001
MNIST-0vs4	**0.992±0.002**	0.991 ± 0.004	0.879±0.031	0.703±0.010	0.933±0.002	0.867±0.010	0.999±0.001
MNIST-0vs5	**0.968±0.026**	0.952 ± 0.010	0.843±0.050	0.704±0.006	0.750±0.002	0.760±0.011	0.986±0.001
MNIST-0vs6	**0.968±0.017**	0.952 ± 0.010	0.847±0.024	0.705±0.008	0.836±0.002	0.806±0.007	0.985±0.004
MNIST-0vs7	**0.986±0.011**	0.979 ± 0.006	0.876±0.020	0.700±0.000	0.953±0.002	0.880±0.007	0.995±0.001
MNIST-0vs8	**0.983±0.005**	0.968 ± 0.006	0.893±0.030	0.703±0.007	0.880±0.002	0.842±0.007	0.989±0.002
MNIST-0vs9	**0.986±0.009**	0.979 ± 0.006	0.902±0.025	0.701±0.002	0.925±0.003	0.860±0.004	0.992±0.001
Fashion-MNIST	**0.925±0.008**	0.900 ± 0.015	0.757±0.034	0.700±0.004	0.678±0.001	0.684±0.002	0.940±0.007

Table 7 Classification accuracy (mean±std) of each method on large-scale datasets by using 25% of the training set to generate similar and dissimilar data pairs

Dataset	OSD-LSPA	OSD-QSPA	OSD-OGD	A-Ramp	KM	CKM	OBC (supervised)
MNIST-0vs1	**0.995±0.004**	0.994 ± 0.004	0.946±0.032	0.700±0.000	0.989±0.001	0.943±0.007	0.999±0.001
MNIST-0vs2	**0.978±0.014**	0.942 ± 0.034	0.878±0.044	0.701±0.008	0.892±0.016	0.839±0.014	0.988±0.002
MNIST-0vs3	**0.977±0.029**	0.953 ± 0.036	0.884±0.050	0.702±0.003	0.873±0.011	0.833±0.009	0.995±0.001
MNIST-0vs4	**0.986±0.012**	0.972 ± 0.017	0.879±0.047	0.703±0.004	0.931±0.009	0.871±0.015	0.998±0.001
MNIST-0vs5	**0.962±0.012**	0.943 ± 0.022	0.832±0.052	0.704±0.006	0.744±0.019	0.686±0.099	0.982±0.002
MNIST-0vs6	**0.956±0.023**	0.922 ± 0.040	0.846±0.055	0.703±0.005	0.842±0.020	0.799±0.027	0.982±0.003
MNIST-0vs7	**0.983±0.009**	0.952 ± 0.021	0.873±0.046	0.700±0.000	0.949±0.006	0.874±0.012	0.992±0.002
MNIST-0vs8	**0.976±0.01**5	0.956 ± 0.020	0.880±0.052	0.701±0.005	0.878±0.012	0.842±0.017	0.987±0.003
MNIST-0vs9	**0.982±0.012**	0.966 ± 0.018	0.885±0.044	0.701±0.005	0.924±0.007	0.858±0.013	0.991±0.001
Fashion-MNIST	**0.920±0.011**	0.885 ± 0.027	0.767±0.042	0.700±0.001	0.678±0.004	0.681±0.005	0.938±0.010

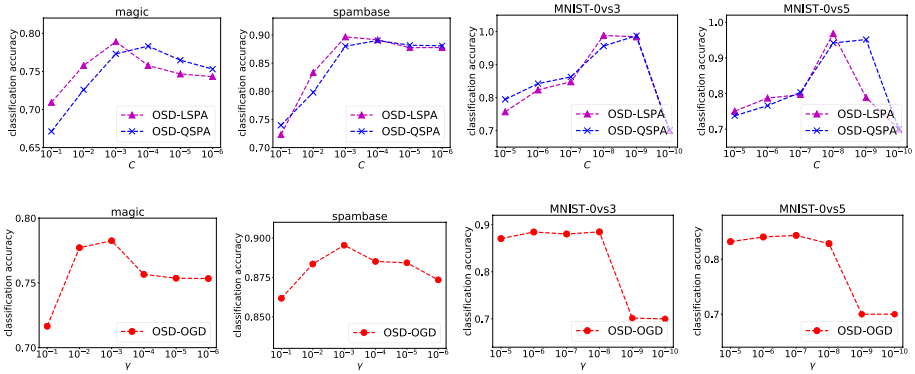

Fig. 1 The sensitivity analysis of parameters C and γ

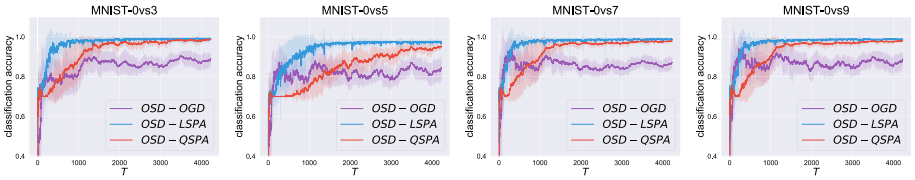

Fig. 2 The classification accuracy of our proposed methods when T increases

5 Conclusion

In this paper, we studied an interesting problem called *online binary classification from only similar and dissimilar data pairs* (online SD classification for short). To the best of our knowledge, this paper provided the first attempt to solve this problem. We proposed three novel learning methods for online SD classification. Specifically, we first adapted the unbiased risk estimator of SD classification to the online learning scenario with a conservative regularization term. Then, by further introducing a margin criterion for whether to update the classifier or not with the received cost, we proposed two online SD classification methods (one with *linearly* scaled cost and the other with *quadratically* scaled cost). We theoretically derived the regret, mistake, and relative loss bounds for our methods, which guarantee the performance of our methods on sequential data. Comprehensive experimental results on various datasets demonstrated the effectiveness of our methods.

There exists a subtle gap between the theoretical analyses and the empirical performance of our proposed methods, due to the estimation error of the class prior. Therefore, in future work, we plan to bridge this gap by incorporating the estimation error of the class prior to the theoretical analyses. In addition, since there exist other important weakly supervised learning problems apart from the SD classification problem, we also plan to develop effective online learning methods for other weakly supervised learning problems in future work.

Appendix 1: Proof of Theorem 1

As we have shown, our proposed OSD-OGD algorithm actually employs the following update method:

$$\boldsymbol{w}_{t+1} = \arg\min_{\boldsymbol{w}\in\mathcal{W}} \sum_{i=1}^{t} R_i^{\mathrm{SD}}(\boldsymbol{w}) + \frac{1}{2\gamma}\|\boldsymbol{w}\|_2^2,$$

which is exactly the follow-the-regularized-leader procedure with Euclidean (Shalev-Shwartz, 2011). As can be easily verified, the Euclidean regularization $\frac{1}{2\gamma}\|\boldsymbol{w}\|_2^2$ is $\frac{1}{\gamma}$ -strongly-convex with respect to $\|\cdot\|_2$. Recall the assumptions that $\mathcal{L}_t$ is ρ_t-Lipschitz with respect to $\|\cdot\|_2$ and $\frac{1}{T}\sum_{t=1}^{T}\rho_t^2 \leq \rho^2$. Then, by using the Theorem 2.11 in Shalev-Shwartz (2011), we have that for $\boldsymbol{w}_\star \in \mathcal{W}$,

$$\sum_{t=1}^{T}\mathcal{L}_t(\boldsymbol{w}_t) - \sum_{t=1}^{T}\mathcal{L}_t(\boldsymbol{w}_\star) \leq \frac{1}{2\gamma}(\|\boldsymbol{w}_\star\|_2^2 - \min_{\boldsymbol{v}\in\mathcal{W}}\|\boldsymbol{v}\|_2^2) + \gamma T\rho^2 \leq \frac{1}{2\gamma}\|\boldsymbol{w}_\star\|_2^2 + \gamma T\rho^2,$$

because $\min_{\boldsymbol{v}\in\mathcal{W}}\|\boldsymbol{v}\|_2^2 \geq 0$ always holds. In particular, if for every hypothesis $\boldsymbol{w} \in \mathcal{W}$, it satisfies $\|\boldsymbol{w}\|_2 \leq B$ and $\gamma = B/(\rho\sqrt{2T})$, we have

$$\sum_{t=1}^{T}\mathcal{L}_t(\boldsymbol{w}_t) - \sum_{t=1}^{T}\mathcal{L}_t(\boldsymbol{w}_\star) \leq B\rho\sqrt{2T},$$

which completes the proof of Theorem 1.

Appendix 2: Proof of Theorem 2

Following (Crammer et al., 2006), for some $\boldsymbol{v} \in \mathcal{W}$, we define

$$\Delta_t = \|\boldsymbol{w}_t - \boldsymbol{v}\|_2^2 - \|\boldsymbol{w}_{t+1} - \boldsymbol{v}\|_2^2$$

and consider upper and lower bounds of $\sum_{t=1}^{T}\Delta_t$. By initializing $\boldsymbol{w}_1$ to zero vector and using telescoping sum, we can obtain

$$\begin{aligned}\sum_{t=1}^{T}\Delta_t &= \sum_{t=1}^{T}(\|\boldsymbol{w}_t - \boldsymbol{v}\|_2^2 - \|\boldsymbol{w}_{t+1} - \boldsymbol{v}\|_2^2)\\ &= \|\boldsymbol{w}_1 - \boldsymbol{v}\|_2^2 - \|\boldsymbol{w}_{T+1} - \boldsymbol{v}\|_2^2\\ &\leq \|\boldsymbol{v}\|_2^2.\end{aligned}$$

Since $\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - \tau_t\nabla R_t^{SD}(\boldsymbol{w})$, we can obtain

$$\begin{aligned}\Delta_t &= \|\boldsymbol{w}_t - \boldsymbol{v}\|_2^2 - \|\boldsymbol{w}_t - \tau_t\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) - \boldsymbol{v}\|_2^2\\ &= 2\tau_t(\boldsymbol{w}_t - \boldsymbol{v})^\top\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) - \tau_t^2\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2.\end{aligned}$$

Since $R_t^{\mathrm{SD}}(\boldsymbol{w})$ is λ-convex, we have

$$R_t^{\mathrm{SD}}(\boldsymbol{v}) - R_t^{\mathrm{SD}}(\boldsymbol{w}_t) \geq (\boldsymbol{v} - \boldsymbol{w}_t)^\top\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t) + \frac{\lambda}{2}\|\boldsymbol{v} - \boldsymbol{w}_t\|_2^2.$$

Combining the above inequalities, we have

$$\|\boldsymbol{v}\|^2 \geq \sum_{t=1}^{T} \tau_t \big(2R_t^{\mathrm{SD}}(\boldsymbol{w}_t) - \tau_t \|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\|_2^2 - 2R_t^{\mathrm{SD}}(\boldsymbol{v})\big). \tag{14}$$

For OSD-LSPA, if a prediction mistake occurs, then $R_t^{\mathrm{SD}}(\boldsymbol{w}_t) \geq G$ and $R_t^{\mathrm{SD}}(\boldsymbol{w}_t) - \tau_t \|\nabla R_t(\boldsymbol{w}_t)\|_2^2 \geq 0$. Therefore, we can obtain

$$\sum_{t=1}^{T} \tau_t R_t^{\mathrm{SD}}(\boldsymbol{w}_t) \leq \|\boldsymbol{v}\|_2^2 + 2C \sum_{t=1}^{T} R_t^{\mathrm{SD}}(\boldsymbol{v}) \tag{15}$$

Using our assumption that $\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\|_2^2 \leq r^2$ and the definitions $\tau_t = \min(C, \max(0, \frac{A + \boldsymbol{w}_t^{\top} \nabla R_t^{\mathrm{SD}}(\boldsymbol{w})}{\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\|_2^2}))$, $R_t^{\mathrm{SD}}(\boldsymbol{w}_t) = A + \boldsymbol{w}_t^{\top} \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)$, we conclude that if a prediction mistake occurs then it holds that

$$\min\left(CG, \frac{G^2}{r^2}\right) \leq \tau_t R_t^{\mathrm{D}}(\boldsymbol{w}_t).$$

Since $\sum_{t=1}^{T} E_t(\boldsymbol{w}_t)$ denote the number of prediction mistakes made on the entire sequence, it holds that

$$\min\left(CG, \frac{G^2}{r^2}\right) \sum_{t=1}^{T} E_t(\boldsymbol{w}_t) \leq \sum_{t=1}^{T} \tau_t R_t^{\mathrm{SD}}(\boldsymbol{w}_t). \tag{16}$$

Combining Eq. (15) with Eq. (16), we conclude that

$$\sum_{t=1}^{T} E_t(\boldsymbol{w}_t) \leq \max\left(\frac{1}{CG}, \frac{r^2}{G^2}\right)\left(\|\boldsymbol{v}\|_2^2 + 2C \sum_{t=1}^{T} R_t^{\mathrm{SD}}(\boldsymbol{v})\right),$$

which completes the proof of Theorem 2.

Appendix 3: Proof of Theorem 3

Recall Eq. (14), we have

$$\|\boldsymbol{v}\|_2^2 \geq \sum_{t=1}^{T} \tau_t \big(2R_t^{\mathrm{SD}}(\boldsymbol{w}_t) - \tau_t \|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\|_2^2 - 2R_t^{\mathrm{SD}}(\boldsymbol{v})\big).$$

Defining $\alpha = 1/\sqrt{2C}$, we subtract the non-negative term $(\alpha\tau_t - R_t^{\mathrm{SD}}(\boldsymbol{v})/\alpha)^2$ from each summand on the right-hand side of the above inequality, to obtain

$$\begin{aligned}
\|\boldsymbol{v}\|_2^2 &\geq \sum_{t=1}^{T} \left(2\tau_t R_t^{\mathrm{SD}}(\boldsymbol{w}_t) - \tau_t^2 \|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\|_2^2 - 2\tau_t R_t^{\mathrm{SD}}(\boldsymbol{v}) - (\alpha\tau_t - R_t^{\mathrm{SD}}(\boldsymbol{v})/\alpha)^2\right) \\
&= \sum_{t=1}^{T} \left(2\tau_t R_t^{\mathrm{SD}}(\boldsymbol{w}_t) - \tau_t^2 \|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\|_2^2 - 2\tau_t R_t^{\mathrm{SD}}(\boldsymbol{v}) - (\alpha\tau_t)^2 \right. \\
&\qquad \left. - \left(\frac{R_t^{\mathrm{SD}}(\boldsymbol{v})}{\alpha}\right)^2 + 2\tau_t R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\right) \\
&= \sum_{t=1}^{T} \left(2\tau_t R_t^{\mathrm{SD}}(\boldsymbol{w}_t) - \tau_t^2 \left(\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w}_t)\|_2^2 + \frac{1}{2C}\right) - 2C(R_t^{\mathrm{SD}}(\boldsymbol{v}))^2\right).
\end{aligned}$$

Using the definitions $\tau_t = \max(0, \frac{2C(A+\boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}))}{2C\|\nabla R_t^{\mathrm{SD}}(\boldsymbol{w})\|_2^2+1})$ and $R_t^{\mathrm{SD}}(\boldsymbol{w}_t) = \boldsymbol{w}_t^\top \nabla R_t^{\mathrm{SD}}(\boldsymbol{w}) + A$. It is clear that when $R_t^{\mathrm{SD}}(\boldsymbol{w}_t) \leq 0$, the classifier is not updated. So we consider the case that $R_t^{\mathrm{SD}}(\boldsymbol{w}_t) \geq 0$. Then we obtain

$$\|\boldsymbol{v}\|_2^2 \geq \sum_{t=1}^{T} \left(\frac{(R_t^{\mathrm{SD}}(\boldsymbol{w}_t))^2}{r^2 + \frac{1}{2C}} - 2C(R_t^{\mathrm{SD}}(\boldsymbol{v}))^2 \right).$$

Rearranging terms above, we can obtain

$$\sum_{t=1}^{T} (R_t^{\mathrm{SD}}(\boldsymbol{w}_t))^2 \leq \left(r^2 + \frac{1}{2C}\right)\left(\|\boldsymbol{v}\|_2^2 + 2C \sum_{t=1}^{T} (R_t^{\mathrm{SD}}(\boldsymbol{v}))^2\right),$$

which completes the proof of Theorem 3.

Author contributions Conceptualization: S-S; methodology: S-S; Theoretical analysis: F-L, H-W; Writing-original draft preparation: S-S, F-L; Writing-review and editing: S-S, Z-W; Funding acquisition: B-H, T-X, B-A, F-L.

Funding This research is supported by Natural Science Foundation of China (No. 62106028), Chongqing Overseas Chinese Entrepreneurship and Innovation Support Program, and CAAI-Huawei MindSpore Open Fund.

Data availability Not applicable.

Code availability Not applicable.

Declarations

Conflict of interest The authors declare that there is no conflict of interest.

Ethics approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

References

Bao, H., Niu, G., & Sugiyama, M. (2018). Classification from pairwise similarity and unlabeled data. In *ICML*, pp. 452–461.

Bao, H., Shimada, T., Xu, L., Sato, I., & Sugiyama, M. (2020). Similarity-based classification: Connecting similarity learning to binary classification. arXiv preprint arXiv:2006.06207.

Blake, C. L., & Merz, C. J. (1998). UCI repository of machine learning databases. http://archive.ics.uci.edu/ml/index.php.

Cao, Y., Feng, L., Xu, Y., An, B., Niu, G., & Sugiyama, M. (2021). Learning from similarity-confidence data. In *ICML*, pp. 1272–1282.

Cao, Y., Wan, Z., Ren, D., Yan, Z., & Zuo, W. (2022). Incorporating semi-supervised and positive-unlabeled learning for boosting full reference image quality assessment. In *CVPR*, pp. 5851–5861.

Chen, R., Tang, Y., Zhang, W., & Feng, W. (2022). Deep multi-view semi-supervised clustering with sample pairwise constraints. *Neurocomputing, 500*, 832–845.

Crammer, K., Kulesza, A., & Dredze, M. (2009). Adaptive regularization of weight vectors. In *NeurIPS*, pp. 414–422.

Crammer, K., & Singer, Y. (2003). Ultraconservative online algorithms for multiclass problems. *Journal of Machine Learning Research, 3*(Jan), 951–991.
Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., & Singer, Y. (2006). Online passive-aggressive algorithms. *Journal of Machine Learning Research, 7*(Mar), 551–585.
Dekel, O., Gilad-Bachrach, R., Shamir, O., & Xiao, L. (2012). Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research*, *13*(1).
Er, M. J., Venkatesan, R., & Wang, N. (2016). An online universal classifier for binary, multi-class and multi-label classification. In *ICSMC*, pp. 003701–003706. IEEE.
Feng, L., Lv, J.-Q., Han, B., Xu, M., Niu, G., Geng, X., An, B., & Sugiyama, M. (2020). Provably consistent partial-label learning. In *NeurIPS*.
Freund, Y., & Schapire, R. E. (1999). Large margin classification using the perceptron algorithm. *Machine Learning, 37*(3), 277–296.
Hoi, S. C., Sahoo, D., Lu, J., & Zhao, P. (2021). Online learning: A comprehensive survey. *Neurocomputing, 459*, 249–289.
Ishida, T., Niu, G., & Sugiyama, M. (2018). Binary classification for positive-confidence data. In *NeurIPS*, pp. 5917–5928.
Jian, L., Gao, F., Ren, P., Song, Y., & Luo, S. (2018). A noise-resilient online learning algorithm for scene classification. *Remote Sensing, 10*(11), 1836.
Kaneko, T., Sato, I., & Sugiyama, M. (2019). Online multiclass classification based on prediction margin for partial feedback. arXiv preprint arXiv:1902.01056.
Kiryo, R., Niu, G., Du Plessis, M. C., & Sugiyama, M. (2017). Positive-unlabeled learning with non-negative risk estimator. In *NeurIPS*, pp. 1675–1685.
Kivinen, J., Smola, A. J., & Williamson, R. C. (2004). Online learning with kernels. *IEEE Transactions on Signal Processing, 52*(8), 2165–2176.
Koçak, M. A., Shasha, D. E., & Erkip, E. (2016). Conjugate conformal prediction for online binary classification. In *UAI*. Citeseer.
LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.
Li, Z., & Liu, J. (2009). Constrained clustering by spectral kernel learning. In *ICCV*, pp. 421–427.
Li, M., Zhang, T., Chen, Y., & Smola, A. J. (2014). Efficient mini-batch training for stochastic optimization. In *KDD*, pp. 661–670.
Liu, D., Zhang, P., & Zheng, Q. (2015). An efficient online active learning algorithm for binary classification. *Pattern Recognition Letters, 68*, 22–26.
Lu, N., Niu, G., Menon, A. K., & Sugiyama, M. (2019). On the minimal supervision for training any binary classifier from only unlabeled data. In *ICLR*.
Lu, N., Zhang, T., Niu, G., & Sugiyama, M. (2020). Mitigating overfitting in supervised classification from two unlabeled datasets: A consistent risk correction approach. In *AISTATS*, pp. 1115–1125.
Lu, D., & Weng, Q. (2007). A survey of image classification methods and techniques for improving classification performance. *International Journal of Remote Sensing, 28*(5), 823–870.
MacQueen, J., et al. (1967). Some methods for classification and analysis of multivariate observations. In *Berkeley symposium on mathematical statistics and probability*, pp. 281–297.
Maheshwara, S. S., & Manwani, N. (2023). Rolnip: Robust learning using noisy pairwise comparisons. In *ACML*, pp. 706–721.
Natarajan, N., Dhillon, I. S., Ravikumar, P. K., & Tewari, A. (2013). Learning with noisy labels. In *NeurIPS*, pp. 1196–1204.
Plessis, M. C., Niu, G., & Sugiyama, M. (2015). Convex formulation for learning from positive and unlabeled data. In *ICML*, pp. 1386–1394.
Shalev-Shwartz, S., et al. (2011). Online learning and online convex optimization. *Foundations and Trends in Machine Learning, 4*(2), 107–194.
Shimada, T., Bao, H., Sato, I., & Sugiyama, M. (2020). Classification from pairwise similarities/dissimilarities and unlabeled data via empirical risk minimization. *Neural Computation*.
Shinoda, K., Kaji, H., & Sugiyama, M. (2020). Binary classification from positive data with skewed confidence. In *IJCAI*, pp. 3328–3334.
Tao, Q., Scott, S., Vinodchandran, N., & Osugi, T. T. (2004). Svm-based generalized multiple-instance learning via approximate box counting. In *ICML*, p. 101.
Wagstaff, K., Cardie, C., Rogers, S., Schrödl, S., et al. (2001). Constrained k-means clustering with background knowledge. In *ICML*, pp. 577–584.
Wang, H., Qiang, Y., Chen, C., Liu, W., Hu, T., Li, Z., & Chen, G. (2020). Online partial label learning. In *ECML PKDD*.

Wu, D.-D., Wang, D.-B., & Zhang, M.-L. (2022). Revisiting consistency regularization for deep partial label learning. In *ICML*, pp. 24212–24225.
Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms. arXiv preprint arXiv:1708.07747.
Xu, D., & Tian, Y. (2015). A comprehensive survey of clustering algorithms. *Annals of Data Science, 2*(2), 165–193.
Zhang, C., Gong, C., Liu, T., Lu, X., Wang, W., & Yang, J. (2020). Online positive and unlabeled learning. In *IJCAI*, pp. 2248–2254.

Authors and Affiliations

Senlin Shu[1] · Haobo Wang[2] · Zhuowei Wang[3] · Bo Han[4] · Tao Xiang[1] · Bo An[5] · Lei Feng[5]

✉ Lei Feng
lfengqaq@gmail.com

Senlin Shu
senlinshu@stu.cqu.edu.cn

Haobo Wang
wanghaobo@zju.edu.cn

Zhuowei Wang
12952560@student.uts.edu.au

Bo Han
bhanml@comp.hkbu.edu.hk

Tao Xiang
txiang@cqu.edu.cn

Bo An
boan@ntu.edu.sg

[1] Chongqing University, Chongqing, China

[2] Zhejiang University, Hangzhou, China

[3] University of Technology Sydney, Sydney, Australia

[4] Hong Kong Baptist University, Hong Kong, China

[5] Nanyang Technological University, Singapore, Singapore