AUC MAXIMIZATION FROM BIASED POSITIVE-UNLABELED DATA WITH CONFIDENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Maximizing the area under the receiver operating characteristic curve (AUC) is a standard approach to imbalanced binary classification. Although positive and negative data are required for maximizing the AUC, negative data are often difficult to collect in some real-world applications due to privacy concerns or the need for specialized expertise to annotate them. Thus, AUC maximization from positive and unlabeled (PU) data has been attracting attention. Existing methods assume that labeled positive data are unbiased samples from the true positive distribution. However, this ideal assumption is often violated in practice. In this paper, we propose a method to maximize the AUC from biased PU data. To address the bias, our key idea is to exploit *confidence*, i.e., the probability that an instance is positive, associated with the small number of labeled positive data. We derive an estimator of the AUC risk using biased PU data with confidence, enabling AUC maximization under such bias. We further show that the rewritten AUC risk induces a Bayes-optimal AUC ranking even when the available confidence is any strictly increasing transformation of the true posterior probability. We experimentally show the effectiveness of our method on eight real-world datasets.

1 INTRODUCTION

In many real-world binary classification tasks such as cyber security (Mirsky et al., 2018; Bagui & Li, 2021), medical care (Yang et al., 2021), product inspection (Park et al., 2016), and fraud detection (Su et al., 2021), *class-imbalance* frequently arises, where the amount of positive data is much smaller than that of negative data (Johnson & Khoshgoftaar, 2019). For such imbalanced data, classification accuracy, which is the standard performance metric in ordinary classification, is not a suitable measure (Yang & Ying, 2022). Instead, the area under the receiver operating characteristic curve (AUC) is widely used (Bradley, 1997; McDermott et al., 2024). The AUC represents the probability that a classifier will rank a positive instance higher than a negative one (Yang & Ying, 2022). Owing to the nature of the ranking, the AUC can adequately measure the classifier's performance even with imbalanced data. Consequently, maximizing the AUC enables accurate classifiers to be learned from imbalanced data (Liu et al., 2020; Yuan et al., 2021a; Yang & Ying, 2022).

Although labeled positive and negative (PN) data are required for AUC maximization, negative data are often difficult to collect in practice. For example, in cyber security, malicious data (positive data) can be collected from public blocklists, but benign data of legitimate users (negative data) are often unavailable due to privacy concerns, and identifying benign data from given unlabeled data requires a high level of expertise (Mirsky et al., 2018; Kumagai et al., 2024; 2025a). In medical care, although positive patient data can be collected on the basis of confirmed diagnoses, treating all other data as negative is problematic since such data often include false-negatives or pre-symptomatic patients who have already contracted the disease (Bekker & Davis, 2020). Similar situations occur in other applications such as product inspection (Kumagai et al., 2024) and fraud detection (Su et al., 2021).

To address such situations, several studies have proposed methods to maximize the AUC from only positive and unlabeled (PU) data (Sakai et al., 2018; Xie & Li, 2018; Xie et al., 2024). These methods assume that labeled positive data are unbiased samples from the true positive distribution, which is known as the *selected completely at random* (SCAR) setting (Elkan & Noto, 2008). Although this ideal assumption enables an unbiased estimator of the AUC to be derived with PU data, it is often violated in practice since labeled positive data are often *biased*. For example, blocklists tend

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

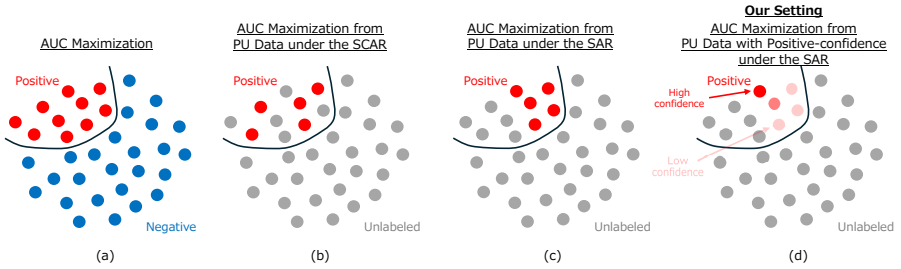


Figure 1: Illustrations of our problem setting and other related settings. Red, blue, and gray points are positive, negative, and unlabeled data, respectively. The dark/light red colors show high/low confidence values for positive data. The black line is illustrated to visually distinguish positive and negative data. (a) Standard AUC maximization with PN data. (b) AUC Maximization from PU data under the SCAR setting where labeled positive data are unbiased. (c) AUC Maximization from PU data under the SAR setting where labeled positive data are biased. Note that there are no studies for this setting. (d) Our problem setting: AUC maximization from PU data with positive-confidence under the SAR setting. In class-imbalanced settings, the number of labeled positive instances (each equipped with confidence) is small.

to contain certain (widely known) types of malicious data, and diseases that can be diagnosed are usually limited to well-understood cases.

To cope with such biased situations, some PU learning studies for ordinary classification have recently considered the *selected at random* (SAR) setting, where the probability of selecting positive data to be labeled depends on their feature values (Kato et al., 2019; Teisseyre et al., 2025; Gerych et al., 2022). While this setting is more realistic, these methods are not designed for maximizing the AUC. In addition, to make the problem tractable, they require strong assumptions about data distributions or labeling mechanisms, such as the PN distributions being separated (Gerych et al., 2022) or the selection probability of positive data following some restricted function forms (Gerych et al., 2022; He et al., 2018; Teisseyre et al., 2025). When the assumption is violated, the performance of these methods drastically deteriorates.

To address this problem, we propose a method for maximizing the AUC under the SAR setting by using biased positive data equipped with *confidence* and unlabeled data, without such restrictive assumptions about data distributions or labeling mechanisms. Here, the confidence is formally defined as the probability that a given instance is positive, and we refer to this confidence for labeled positive data as positive-confidence. Figure 1 illustrates our problem setting.

Such confidence need not be collected specifically for the proposed method; in many applications, it is already produced as a by-product of existing labeling or operational processes. For example, when the same instance is labeled by multiple annotators to enhance reliability, the average of these labels can naturally form confidence (Ishida et al., 2023). Such multi-annotator labeling is used in various domains such as medical care, cyber security, and crowdsourcing (Le et al., 2023; Kantchelian et al., 2015; Tang et al., 2025). Confidence-like scores are also routinely available in applications, such as BI-RADS scores representing the likelihood of cancer in medical care (Sato, 2025), alert severity levels in security operations (Jalalvand et al., 2024), and risk scores in financial fraud detection. Moreover, automatic labeling systems, including LLM-based annotators and pre-trained classifiers, can also provide such information (Ishida et al., 2018; Tang et al., 2025). These examples motivate the growing body of confidence-based weakly supervised learning methods (Ishida et al., 2018; Cao et al., 2021; Berthon et al., 2021; Feng et al., 2021; Wang et al., 2023b).

However, to the best of our knowledge, positive-confidence has not been used for PU learning under the SAR setting. The key idea is that positive-confidence provides information that allows us to correct the bias in the labeled positive data. By using such confidence, we derive an estimator of the AUC risk from biased PU data without relying on the strong assumptions about data distributions or labeling mechanisms required by existing SAR methods. This enables us to maximize the AUC from biased PU data. A practical concern is that obtained confidence may not be accurately calibrated as a probability. We further show that the proposed method does not require exact calibration to recover the optimal AUC ranking: when the observed confidence is given by any strictly increasing

transformation of the true posterior probability[1], minimizing the resulting AUC risk still yields the Bayes-optimal AUC ranking. This indicates that the proposed method can also exploit confidence that is systematically over- or under-confident, as long as its ordering is preserved. In addition, our method is model-agnostic: it can use any differentiable classifier such as linear classifier, kernel-based classifier, and neural network, which is preferable in practice.

2 Related Work

Various methods for AUC maximization have been proposed (Brefeld et al., 2005; Ying et al., 2016; Liu et al., 2020; Yuan et al., 2021a; Yang & Ying, 2022). Previous studies (Fujino & Ueda, 2016; Yuan et al., 2021a;b; Wang et al., 2023a) have shown that AUC maximization methods often outperform other methods for imbalanced data such as class balanced loss (Charoenphakdee et al., 2019), focal loss (Lin et al., 2017), or sampling-based methods (Menardi & Torelli, 2014; Chawla et al., 2002). However, these AUC maximization methods require both labeled PN data, and thus, they cannot be applied to our problem setting.

PU learning methods train binary classifiers by using only PU data (Bekker & Davis, 2020). A representative approach is empirical risk minimization, which rewrites the empirical risk by using only PU data (Sugiyama et al., 2022). Although most methods are designed for ordinary classification on balanced data (Du Plessis et al., 2015; Kiryo et al., 2017; Jiang et al., 2023), several studies have recently shown the AUC can also be maximized from PU data (Sakai et al., 2018; Xie & Li, 2018; Charoenphakdee et al., 2019; Xie et al., 2024). However, these PU learning and AUC maximization methods assume the SCAR setting, i.e., labeled positive data are unbiased, which is not often satisfied in practice. This paper tackles the more realistic but challenging SAR setting where labeled positive data are biased.

Some PU learning methods for the SAR setting have recently been proposed (Bekker et al., 2019; Kato et al., 2019; Gerych et al., 2022; He et al., 2018; Teisseyre et al., 2025). Although several studies used EM algorithm-based methods (Bekker et al., 2019; Gong et al., 2021), they lack theoretical guarantees for learning the true classifier (Gerych et al., 2022; Teisseyre et al., 2025). To make the problem theoretically tractable, most studies impose restrictive assumptions about data distributions or labeling mechanisms, such as the data separation assumption (Gerych et al., 2022), the invariance of order assumption (Kato et al., 2019), and the probabilistic gap assumption (He et al., 2018; Gerych et al., 2022). However, when the assumption is violated, they cannot work well. In addition, they are not designed for maximizing the AUC. In contrast, the proposed method can maximize the AUC from biased PU data using positive-confidence, without relying on such restrictive assumptions.

Several confidence-based learning methods have recently been proposed (Ishida et al., 2018; Cao et al., 2021; Berthon et al., 2021; Feng et al., 2021; Wang et al., 2023b). Especially, Ishida et al. (2018) and Shinoda et al. (2021) use positive-confidence to learn binary classifiers. Although they do not require unlabeled data, they cannot be applied for maximizing the AUC, and require that labeled positive data are unbiased. One PU learning method uses positive-confidence information to deal with noisy labels (Tang et al., 2025). However, it is not designed for either AUC maximization or the SAR setting. In addition, existing confidence-based learning methods typically formulate confidence as true posterior probabilities. In contrast, the proposed method can maximize the AUC even if the observed confidence is given by any strictly increasing transformation of the true posterior.

3 Preliminaries

We briefly introduce AUC maximization. Let input instance $\mathbf{x} \in \mathcal{X}$ and its class label $y \in \{0, 1\}$ be associated with probability density $p(\mathbf{x}, y)$, where 1 and 0 correspond to PN classes, respectively. The conditional probability densities for PN classes are denoted as $p^{\mathrm{p}}(\mathbf{x}) := p(\mathbf{x}|y = 1)$ and $p^{\mathrm{n}}(\mathbf{x}) := p(\mathbf{x}|y = 0)$. Let $s : \mathcal{X} \to \mathbb{R}$ be a score function that quantifies how likely an instance is positive. The classifier is defined by the score function with threshold t: $\hat{y} = I(s(\mathbf{x}) \geq t)$, where $I(z)$ is the indicator function that outputs 1 if z is true and 0 otherwise.

The AUC is the probability that a randomly drawn positive instance is ranked higher than a randomly drawn negative instance with ties counted as one half (Yang & Ying, 2022). Formally, the AUC with

[1]We say that a function $h : [0, 1] \to [0, 1]$ is strictly increasing if it satisfies $h(a) < h(b)$ whenever $a < b$.

score function s can be represented as

$$\begin{aligned}\mathrm{AUC}(s) &= \mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{p}}(\mathbf{x})}\mathbb{E}_{\mathbf{x}^{\mathrm{n}}\sim p^{\mathrm{n}}(\mathbf{x})}\left[\ell_{01}(s(\mathbf{x}^{\mathrm{n}}) - s(\mathbf{x}^{\mathrm{p}}))\right] \\ &= 1 - \mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{p}}(\mathbf{x})}\mathbb{E}_{\mathbf{x}^{\mathrm{n}}\sim p^{\mathrm{n}}(\mathbf{x})}\left[\ell_{01}(s(\mathbf{x}^{\mathrm{p}}) - s(\mathbf{x}^{\mathrm{n}}))\right], \end{aligned} \tag{1}$$

where $\ell_{01}(z)$ is the zero-one loss, with $\ell_{01}(z) = 1$ if $z < 0$, 0 if $z > 0$, and $1/2$ if $z = 0$, and $\mathbb{E}$ denotes the expectation. Maximizing the AUC is equivalent to minimizing the following AUC risk,

$$\mathcal{R}(s) := \mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{p}}(\mathbf{x})}\mathbb{E}_{\mathbf{x}^{\mathrm{n}}\sim p^{\mathrm{n}}(\mathbf{x})}\left[\ell_{01}(s(\mathbf{x}^{\mathrm{p}}) - s(\mathbf{x}^{\mathrm{n}}))\right]. \tag{2}$$

Since the gradient of the zero-one loss is zero almost everywhere, the AUC risk cannot be directly minimized with gradient descent methods. To address this, a common approach is to use the smoothed AUC risk by replacing $\ell_{01}(z)$ with the sigmoid surrogate $\sigma(-z)$, where $\sigma(z) = 1/(1+\exp(-z))$ (Charoenphakdee et al., 2019):

$$\mathcal{R}_{\sigma}(s) := \mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{p}}(\mathbf{x})}\mathbb{E}_{\mathbf{x}^{\mathrm{n}}\sim p^{\mathrm{n}}(\mathbf{x})}\left[\sigma(-s(\mathbf{x}^{\mathrm{p}}) + s(\mathbf{x}^{\mathrm{n}}))\right]. \tag{3}$$

Suppose we have N^{p} positive instances $\{\mathbf{x}^{\mathrm{p}}_1, \ldots, \mathbf{x}^{\mathrm{p}}_{N^{\mathrm{p}}}\}$ drawn from $p^{\mathrm{p}}(\mathbf{x})$ and N^{n} negative instances $\{\mathbf{x}^{\mathrm{n}}_1, \ldots, \mathbf{x}^{\mathrm{n}}_{N^{\mathrm{n}}}\}$ drawn from $p^{\mathrm{n}}(\mathbf{x})$, the empirical estimator of Eq. (3) is given as

$$\widehat{\mathcal{R}}_{\sigma}(s) = \frac{1}{N^{\mathrm{p}}N^{\mathrm{n}}}\sum_{n=1}^{N^{\mathrm{p}}}\sum_{m=1}^{N^{\mathrm{n}}}\sigma(-s(\mathbf{x}^{\mathrm{p}}_n) + s(\mathbf{x}^{\mathrm{n}}_m)). \tag{4}$$

By minimizing this empirical AUC risk w.r.t. the parameters of s, we can obtain good score functions to maximize the AUC (Yang & Ying, 2022).

4 Proposed Method

4.1 Problem Formulation

We assume that there exists probability density $p(\mathbf{x}, y, o)$, where $\mathbf{x} \in \mathcal{X}$ is an input instance, $y \in \{0, 1\}$ is its class, and $o \in \{0, 1\}$ represents whether its class y is observed (labeled). Here, $o = 1$ denotes that its class is labeled, and $o = 0$ denotes that its class is unlabeled. In PU learning setting, only positive data are labeled. This PU assumption can be formally represented as

$$p(o = 1|\mathbf{x}, y = 0) = 0, \tag{5}$$

which means that negative data are never labeled (Elkan & Noto, 2008). Propensity score $e(\mathbf{x})$ is defined as

$$e(\mathbf{x}) := p(o = 1|\mathbf{x}, y = 1), \tag{6}$$

which represents the probability that positive instance $\mathbf{x}$ is labeled. Most PU learning methods assume the SCAR setting, i.e., propensity score $e(\mathbf{x})$ does not depend on feature values $\mathbf{x}$, $e(\mathbf{x}) = p(o = 1|y = 1)$ (Sakai et al., 2018; Kiryo et al., 2017; Xie & Li, 2018; Xie et al., 2024). Although this assumption simplifies the problem, it is seldom met in practice. Thus, this paper assumes the SAR setting,

$$e(\mathbf{x}) = p(o = 1|\mathbf{x}, y = 1) \not\equiv p(o = 1|y = 1), \tag{7}$$

which means that $e(\mathbf{x})$ depends on feature values $\mathbf{x}$. This is a more realistic but difficult setting. Following previous studies on biased PU learning (Bekker et al., 2019; Bekker & Davis, 2020; Gerych et al., 2022), we assume the positivity condition that $e(\mathbf{x}) > 0$ for p^{p}-almost every $\mathbf{x}$.

For collecting PU data, there are two possible scenarios: the *one-sample scenario* (also known as the *censoring scenario*) (Elkan & Noto, 2008) and the *two-sample scenario* (also known as the *case-control scenario*) (Ward et al., 2009). This paper focuses on the one-sample scenario since it is more commonly used in PU learning studies (Bekker & Davis, 2020). We can easily modify the proposed method for the two-sample scenario, which is described in Section D.2. In the one-sample scenario, a set of unlabeled data is first sampled from true marginal density $p(\mathbf{x}) = \pi p^{\mathrm{p}}(\mathbf{x}) + (1 - \pi)p^{\mathrm{n}}(\mathbf{x})$ where $\pi := p(y = 1)$ is the positive class-prior. Then, if instance $\mathbf{x}$ is positive, it can be labeled with probability $e(\mathbf{x})$, and $\mathbf{x}$ remains unlabeled with probability $1 - e(\mathbf{x})$; if instance $\mathbf{x}$ is negative, it is never labeled and thus $\mathbf{x}$ remains unlabeled with probability 1. In our problem setting, we

additionally assume that each labeled positive instance $\mathbf{x}$ is equipped with confidence $r(\mathbf{x}) := p(y = 1|\mathbf{x})$. Note that this equality does not have to strictly hold as shown later. As a result, we are given PU data $X^{\mathrm{p}} \cup X$ with positive-confidence R^{p}:

$$X^{\mathrm{p}} := \{\mathbf{x}_n^{\mathrm{p}}\}_{n=1}^{N^{\mathrm{p}}} \sim p^{\mathrm{l}}(\mathbf{x}) := p(\mathbf{x}|o = 1), \;\; X := \{\mathbf{x}_m\}_{m=1}^{N} \sim p^{\mathrm{u}}(\mathbf{x}) := p(\mathbf{x}|o = 0),$$
$$R^{\mathrm{p}} := \{r_n^{\mathrm{p}}|\, r_n^{\mathrm{p}} = p(y = 1|\mathbf{x}_n^{\mathrm{p}})\}_{n=1}^{N^{\mathrm{p}}}, \tag{8}$$

where $p^{\mathrm{l}}(\mathbf{x})$ and $p^{\mathrm{u}}(\mathbf{x})$ are labeled (positive) and unlabeled densities, respectively. From the data generation process, true marginal density $p(\mathbf{x})$ can be also represented as $p(\mathbf{x}) = \alpha p^{\mathrm{l}}(\mathbf{x}) + (1 - \alpha)p^{\mathrm{u}}(\mathbf{x})$, where $c := p(o = 1|y = 1)$ and $\alpha := p(o = 1) = c\pi$[2]. Note that $p^{\mathrm{l}}(\mathbf{x}) \neq p^{\mathrm{p}}(\mathbf{x})$ in the SAR setting. Our goal is to learn score function s that can maximize the AUC under the SAR setting by using given PU data with positive-confidence $X^{\mathrm{p}} \cup X \cup R^{\mathrm{p}}$.

4.2 AUC Risk Estimator with Biased PU Data and Positive-confidence

We derive the AUC risk estimator in the SAR setting from biased PU data with positive-confidence. The objective function to be minimized is the following smoothed AUC risk,

$$\mathcal{R}_\sigma(s) = \mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \mathbb{E}_{\mathbf{x}^{\mathrm{n}} \sim p^{\mathrm{n}}(\mathbf{x})} \left[f(\mathbf{x}^{\mathrm{p}}, \mathbf{x}^{\mathrm{n}}) \right], \tag{9}$$

where we set $f(\mathbf{x}^{\mathrm{p}}, \mathbf{x}^{\mathrm{n}}) := \sigma(-s(\mathbf{x}^{\mathrm{p}}) + s(\mathbf{x}^{\mathrm{n}}))$. Although this AUC risk depends on $p^{\mathrm{p}}(\mathbf{x})$ and $p^{\mathrm{n}}(\mathbf{x})$, data from both the distributions are unavailable in our setting; it seems impossible to calculate. However, we show that it is in fact possible below.

First, by the definition $p(\mathbf{x}) = \pi p^{\mathrm{p}}(\mathbf{x}) + (1 - \pi)p^{\mathrm{n}}(\mathbf{x})$, the negative density can be represented as

$$p^{\mathrm{n}}(\mathbf{x}) = \frac{1}{1 - \pi} \left[p(\mathbf{x}) - \pi p^{\mathrm{p}}(\mathbf{x}) \right]. \tag{10}$$

By substituting Eq. (10) into Eq. (9), we obtain

$$\mathcal{R}_\sigma(s) = \frac{1}{1 - \pi} \mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \left[f(\mathbf{x}^{\mathrm{p}}, \mathbf{x}) \right] - \frac{\pi}{1 - \pi} \mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \left[f(\mathbf{x}^{\mathrm{p}}, \bar{\mathbf{x}}^{\mathrm{p}}) \right]. \tag{11}$$

Here, $\mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \left[f(\mathbf{x}^{\mathrm{p}}, \bar{\mathbf{x}}^{\mathrm{p}}) \right] = 1/2$ due to the symmetry of sigmoid function σ (Xie & Li, 2018; Charoenphakdee et al., 2019; Xie et al., 2024) (The proof is described in Section C.1). Thus, the AUC risk can be represented as

$$\mathcal{R}_\sigma(s) = \frac{1}{1 - \pi} \mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \left[f(\mathbf{x}^{\mathrm{p}}, \mathbf{x}) \right] - \frac{\pi}{2(1 - \pi)}. \tag{12}$$

Labeled density $p^{\mathrm{l}}(\mathbf{x})$ and positive density $p^{\mathrm{p}}(\mathbf{x})$ are related as follows,

$$p^{\mathrm{l}}(\mathbf{x}) = \frac{e(\mathbf{x})}{c} p^{\mathrm{p}}(\mathbf{x}), \tag{13}$$

which means that a positive instance is first sampled from $p^{\mathrm{p}}(\mathbf{x})$ and then is observed as labeled data with probability in accordance with propensity score $e(\mathbf{x})$ (Bekker & Davis, 2020). The derivation of Eq. (13) is described in Section C.2. Under the positivity condition described in Section 4.1, Eq. (13) can be inverted as $p^{\mathrm{p}}(\mathbf{x}) = cp^{\mathrm{l}}(\mathbf{x})/e(\mathbf{x})$ for p^{p}-almost every $\mathbf{x}$.

Additionally, by the PU assumption, $p(o = 1|\mathbf{x}) = e(\mathbf{x})p(y = 1|\mathbf{x})$, as shown in Section C.3. Thus, for $p(y = 1|\mathbf{x}) > 0$, we obtain

$$e(\mathbf{x}) = \frac{p(o = 1|\mathbf{x})}{p(y = 1|\mathbf{x})}. \tag{14}$$

Since $p^{\mathrm{p}}(\mathbf{x}) = p(y = 1|\mathbf{x})p(\mathbf{x})/\pi$, we have $p(y = 1|\mathbf{x}) > 0$ for p^{p}-almost every $\mathbf{x}$. Thus, Eq. (14) can be used together with the inverted form of Eq. (13) in the p^{p}-expectation in Eq. (12).[3] We can therefore rewrite Eq. (12) as

$$\mathcal{R}_\sigma(s) = \frac{c}{1 - \pi} \mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{l}}(\mathbf{x})} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \left[\frac{p(y = 1|\mathbf{x}^{\mathrm{p}})}{p(o = 1|\mathbf{x}^{\mathrm{p}})} f(\mathbf{x}^{\mathrm{p}}, \mathbf{x}) \right] - \frac{\pi}{2(1 - \pi)}. \tag{15}$$

[2] $\alpha = p(o = 1) = p(o = 1|y = 1)p(y = 1) + p(o = 1|y = 0)p(y = 0) = p(o = 1|y = 1)p(y = 1) = c\pi$, where we used the PU property (negative data are never labeled, i.e., $p(o = 1|y = 0) = 0$).

[3] The denominator in the resulting expectation is also nonzero almost surely. In fact, since $p^{\mathrm{l}}(\mathbf{x}) = p(o = 1|\mathbf{x})p(\mathbf{x})/p(o = 1)$, we have $p(o = 1|\mathbf{x}) > 0$ for p^{l}-almost every $\mathbf{x}$.

From this equation, we can see that weight $p(y=1|\mathbf{x}^{\mathrm{p}})/p(o=1|\mathbf{x}^{\mathrm{p}})$ becomes larger for labeled positive instance $\mathbf{x}^{\mathrm{p}}$ with higher $p(y=1|\mathbf{x}^{\mathrm{p}})$ (i.e., more likely to be positive) and lower $p(o=1|\mathbf{x}^{\mathrm{p}})$ (i.e., rarer as labeled data). By substituting $p(\mathbf{x}) = \alpha p^{\mathrm{l}}(\mathbf{x}) + (1-\alpha)p^{\mathrm{u}}(\mathbf{x})$ into Eq. (15), we obtain

$$\mathcal{R}_\sigma(s) = \frac{c}{1-\pi}\left[\alpha\mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\frac{p(y=1|\mathbf{x}^{\mathrm{p}})}{p(o=1|\mathbf{x}^{\mathrm{p}})}f(\mathbf{x}^{\mathrm{p}},\bar{\mathbf{x}}^{\mathrm{p}})\right]\right.$$
$$\left. + (1-\alpha)\mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\mathbb{E}_{\mathbf{x}\sim p^{\mathrm{u}}(\mathbf{x})}\left[\frac{p(y=1|\mathbf{x}^{\mathrm{p}})}{p(o=1|\mathbf{x}^{\mathrm{p}})}f(\mathbf{x}^{\mathrm{p}},\mathbf{x})\right]\right] - \frac{\pi}{2(1-\pi)}. \tag{16}$$

Here, we train probabilistic classifier $\hat{u}(\mathbf{x})$ to estimate $u(\mathbf{x}) := p(o=1|\mathbf{x})$ using given training data $X^{\mathrm{p}} \cup X$. We also estimate labeled probability $\alpha = p(o=1)$ as $\hat{\alpha} := N^{\mathrm{p}}/(N^{\mathrm{p}}+N)$. In addition, the confidence for labeled positive instances, $r_n^{\mathrm{p}} = p(y=1|\mathbf{x}_n^{\mathrm{p}})$, is given in our setting. Thus, by replacing $u(\mathbf{x}) = p(o=1|\mathbf{x})$ and α with their estimates and each expectation with the sample average in Eq. (16), we obtain the following estimator of the AUC risk:

$$\hat{\mathcal{R}}_\sigma(s) = \frac{c}{1-\pi}\left[\frac{\hat{\alpha}}{N^{\mathrm{p}}(N^{\mathrm{p}}-1)}\sum_{n\neq m}^{N^{\mathrm{p}},N^{\mathrm{p}}}\frac{r_n^{\mathrm{p}}}{\hat{u}_n^{\mathrm{p}}}f(\mathbf{x}_n^{\mathrm{p}},\bar{\mathbf{x}}_m^{\mathrm{p}}) + \frac{(1-\hat{\alpha})}{N^{\mathrm{p}}N}\sum_{n,m=1}^{N^{\mathrm{p}},N}\frac{r_n^{\mathrm{p}}}{\hat{u}_n^{\mathrm{p}}}f(\mathbf{x}_n^{\mathrm{p}},\mathbf{x}_m)\right] - \frac{\pi}{2(1-\pi)}. \tag{17}$$

where $\hat{u}_n^{\mathrm{p}} := \hat{u}(\mathbf{x}_n^{\mathrm{p}})$, and we omit the case of $n=m$ in the first term to avoid biases following (Sakai et al., 2018). Since coefficient $c/(1-\pi)$ and constant $\pi/(2(1-\pi))$ do not affect the optimization of s, we can safely ignore them for training. This property is preferable in practice since π and c are difficult or impossible to estimate (Bekker & Davis, 2020). Note that although we use the sigmoid function in the AUC risk in Eq. (9), we can derive the AUC risk estimator of the same form in Eq. (17) whenever we use symmetric functions (i.e., function σ satisfying $\sigma(z)+\sigma(-z)=k$ for any $z\in\mathbb{R}$ and k is a constant). This is because the second term in Eq. (11) also becomes constant. The symmetric functions include many common loss functions such as sigmoid, ramp, and unhinged functions (Charoenphakdee et al., 2019). Even when using non-symmetric functions, we can derive a slightly modified version of the AUC risk estimator, which also enables us to maximize the AUC without requiring π and c. The details are described in Section D.1. We further analyze the sensitivity of the rewritten risk to labeling probability estimation errors in Section B.2 and the generalization behavior of its empirical estimator in Section B.3.

Algorithm 1 shows the training procedure of the proposed method with stochastic gradient descent methods. We first set mini-batch sizes U and P so as to maintain the original label ratio (Line 1). Then, we train probabilistic binary classifier $\hat{u}(\mathbf{x})$ to estimate $p(o=1|\mathbf{x})$ with PU data $X^{\mathrm{p}} \cup X$ (Line 2). Then, we randomly sample PU data with positive-confidence from given data $X^{\mathrm{p}} \cup X \cup R^{\mathrm{p}}$ (Lines 4–5). We calculate the loss in Eq. (17) for learning score function s with trained classifier $\hat{u}(\mathbf{x})$ (Line 6). We update parameters of score function s by using the gradient of the loss (Line 7). In the gradient calculation of the last step, parameters of trained classifier $\hat{u}(\mathbf{x})$ are frozen.

4.3 Robustness to Strictly Increasing Transformations of Positive-Confidence

By replacing true posterior $p(y=1|\mathbf{x})$ in Eq. (15) with observed confidence $r(\mathbf{x})$, we obtain the objective, which is exactly equivalent to the original AUC risk when $r(\mathbf{x}) = p(y=1|\mathbf{x})$. However, we show below that this exact equality is not necessary to recover the Bayes-optimal ranking for the AUC.

Proposition 4.1. *Suppose that the available confidence is $r'(\mathbf{x}) = h\left(p(y=1|\mathbf{x})\right)$, where $h : [0,1] \to [0,1]$ is an arbitrary strictly increasing function. Then, the objective obtained by replacing $p(y=1|\mathbf{x})$ with $r'(\mathbf{x})$ in Eq. (15), i.e.,*

$$\mathcal{R}_\sigma^h(s) = \frac{c}{1-\pi}\mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\frac{r'(\mathbf{x}^{\mathrm{p}})}{p(o=1|\mathbf{x}^{\mathrm{p}})}f(\mathbf{x}^{\mathrm{p}},\mathbf{x})\right] - \frac{\pi}{2(1-\pi)} \tag{18}$$

induces the same Bayes-optimal ranking for the AUC as the original AUC risk in Eq. (9).

The proof is provided in Section B.1. This result shows that the proposed method can recover the Bayes-optimal ranking for the AUC even when the available confidence is given as any strictly increasing transformation of the true posterior probability which includes systematic over- or under-confidence. In this sense, the proposed method can treat a broader class of confidence than true posterior probabilities.

Algorithm 1 Training procedure of the proposed method

Require: Biased PU data with positive-confidence $X^{\mathrm{p}} \cup X \cup R^{\mathrm{p}}$ and total mini-batch size M
Ensure: Model parameters of score function s
1: Partition M into unlabeled and positive mini-batch sizes, U and P, so that $P/(P+U) = N^{\mathrm{p}}/(N^{\mathrm{p}}+N) =: \hat{\alpha}$
2: Train classifier $\hat{u}(\mathbf{x})$ to estimate $p(o=1|\mathbf{x})$ with PU data $X^{\mathrm{p}} \cup X$
3: **repeat**
4: Sample unlabeled data with size U from X
5: Sample positive data with size P and their confidence from $X^{\mathrm{p}} \cup R^{\mathrm{p}}$
6: Calculate the loss in Eq. (17) on the sampled data with trained classifier $\hat{u}(\mathbf{x})$
7: Update parameters of s with the gradient of the loss
8: **until** End condition is satisfied;

5 Experiments

5.1 Data

We mainly used six real-world datasets: Mnist (LeCun et al., 1998), FashionMnist (Fmnist) (Xiao et al., 2017), Svhn (Netzer et al., 2011), Cifar10 (Krizhevsky et al., 2009), Diabetes (Gardner et al., 2023), and Blood (Gardner et al., 2023). The first four datasets are image datasets, while the remaining two are tabular datasets. These datasets have been commonly used in PU learning studies (Kumagai et al., 2024; 2025a;b; Xie et al., 2024; Kiryo et al., 2017; Jiang et al., 2023). Following the previous studies (Kumagai et al., 2025b; 2024; 2025a; Xie et al., 2024), we constructed binary classification problems for the image datasets by partitioning their original classes into positive and negative classes, while using the original binary labels for Diabetes and Blood. The detailed construction is provided in Section E.1.

For training and validation in each dataset, we first randomly sampled $5,000$ and $1,000$ instances from the original data with positive class-prior π, respectively. To create the class-imbalanced data, we set π to a small value. Specifically, we changed π within $\{0.05, 0.1, 0.15, 0.2\}$ in our experiments. Then, we set positive labeling rate $c = p(o=1|y=1)$ to 0.1[4]. We then selected a fraction c of the positive instances within the initially sampled data as labeled positive data according to a biased sampling scheme. Specifically, to induce sampling bias, we made instances from a subset of the positive classes more likely to be labeled for the image datasets and introduced a distance-based selection bias for the tabular datasets. The detailed selection procedure for each dataset and results under other selection biases are provided in Sections E.1 and F.4, respectively. We used $2,500$ positive and $2,500$ negative data as test data for evaluation.

In the main experiments, we generated positive-confidence using an independently trained probabilistic classifier, following the standard evaluation protocol in previous confidence-based learning studies (Ishida et al., 2018; Cao et al., 2021; Berthon et al., 2021; Wang et al., 2023b). This protocol enables us to quantitatively assess the effectiveness of our AUC risk estimator. The classifier was trained on a separate labeled set ($10,000$ instances with class-prior π) that did not overlap with the training, validation, or test data. These labeled data were used solely to obtain confidence values for controlled evaluation. Confidence obtained from pre-trained classifiers is also practically relevant, as such classifiers are widely deployed in many real-world systems. We also report results with positive-confidence obtained from real-world human annotators in Section 5.3. All labeled positive data for training and validation were equipped with positive-confidence. We conducted 10 experiments for each positive class-prior π changing the random seeds and evaluated the mean test AUC.

5.2 Comparison Methods

We compared the proposed method (Ours) with eight methods: non-traditional classifier-based method (NTC) (Elkan & Noto, 2008), non-negative PU learning method (nnPU) (Kiryo et al., 2017),

[4]Since the number of labeled positive training data N^{p} is determined as $N^{\mathrm{p}} = Vp(o=1) = Vp(o=1|y=1)p(y=1) = Vc\pi$, where V denotes the number of initially sampled training instances (i.e., $V = 5,000$), N^{p} becomes 25, 50, 75, and 100 for the respective values of π.

Table 1: Average test AUCs over different positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$. We set $c = p(o = 1|y = 1) = 0.1$. Values in bold are not statistically different at the 5% level from the best performing method in each row according to a paired t-test.

Data	Ours	w/oConf	NTC	nnPU	PUAUC	PUSB	PG	Pconf	NPU
Mnist	**0.8979**	0.8769	0.8575	0.8196	0.8828	0.8841	0.8575	0.8193	0.8285
Fmnist	**0.9495**	0.9371	0.9165	0.9296	0.9409	**0.9474**	0.9165	0.7737	**0.9470**
Svhn	**0.6673**	**0.6666**	0.5328	0.4885	**0.6704**	0.5965	0.5328	0.4900	0.4893
Cifar10	**0.8643**	0.8371	0.8013	0.4819	**0.8620**	0.8442	0.8013	0.4767	0.4883
Diabetes	**0.7342**	0.6926	**0.7357**	0.5234	0.7171	0.7230	**0.7357**	0.3974	0.4626
Blood	**0.5509**	0.5287	0.5409	0.4847	0.5395	**0.5450**	0.5409	0.4962	0.4828

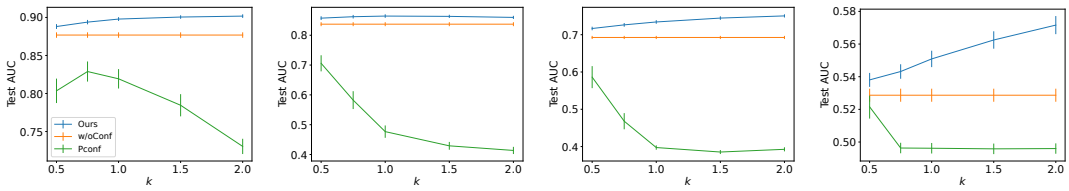
AUC maximization method with PU data (PUAUC) (Xie & Li, 2018; Xie et al., 2024), PU learning method with selection bias (PUSB) (Kato et al., 2019), probabilistic gap-based method (PG) (Gerych et al., 2022), positive-confidence-based method (Pconf) (Ishida et al., 2018), confidence-based noisy PU learning method (NPU) (Tang et al., 2025), and the proposed method without positive-confidence (w/oConf). We further evaluated LBE, a PU learning method under the SAR setting in Section F.6. All methods including our method used neural networks for modeling classifiers.

NTC learns the binary classifier with PU data by simply treating all unlabeled data as negative data. The proposed method also used it for estimating $p(o = 1|\mathbf{x})$. nnPU and PUAUC are PU learning methods for the SCAR setting (i.e., they assume that labeled positive data are unbiased). Specifically, nnPU learns the classifier by minimizing the non-negative PU risk with PU data. PUAUC learns the classifier by minimizing the AUC risk that is calculated with PU data. PUSB and PG are PU learning methods for the SAR setting (i.e., they assume that labeled positive data are biased). To deal with the SAR setting, PUSB assumes that $p(o = 1|\mathbf{x})$ and $p(y = 1|\mathbf{x})$ induce the same order on the feature space. PG assumes propensity score $e(\mathbf{x})$ to be a linear function of $p(y = 1|\mathbf{x})$. PUSB and PG do not use positive-confidence information. In contrast, Pconf and NPU use positive-confidence information as in the proposed method. Pconf learns the classifier by using only positive data with confidence. NPU learns the classifier by using PU data with positive-confidence, which is designed to handle label noise in labeled positive data. w/oConf is the proposed method that assumes $r_n^{\mathrm{p}} = 1$ for all labeled positive data in Eq. (17). We compared it to evaluate the effectiveness of using positive-confidence in our framework. For nnPU, PUSB, and NPU, the absolute loss correction was used for preventing overfitting (Kumagai et al., 2025b).

For the proposed method and w/oConf, we rounded up $\hat{u}_n^{\mathrm{p}} = \hat{u}(\mathbf{x}_n^{\mathrm{p}})$ in Eq. (17) less than 0.01 to 0.01 to stabilize the training process following the previous study (Ishida et al., 2018). We empirically confirm the robustness to this clipping value in Section F.9. The empirical risk with validation PU data was used for early-stopping to mitigate overfitting. All methods were implemented using PyTorch (Paszke et al., 2017). The details of settings such as network architectures are described in Section E.3.

5.3 Results

Table 1 shows the average test AUCs of each method on the six real-world datasets. Full results including the standard deviations are described in Section F.11. The proposed method performed the best or comparably to it in all cases. NTC tended not to work well since it treats all unlabeled data as negative, which leads to biased results. nnPU and PUAUC did not work well since they are not designed for the SAR setting. In particular, since the main difference between the proposed method and PUAUC is whether the SAR setting is taken into account or not, this result indicates the importance of considering the bias of labeled positive data in AUC maximization framework. PUSB and PG, which are PU learning methods for the SAR setting, tended to perform worse than the proposed method because their assumptions to deal with the SAR setting were invalid in our experiments. By using positive-confidence information, the proposed method was able to perform well without relying on such assumptions. Since Pconf does not use unlabeled data and assumes that labeled positive data are unbiased, it also did not work well even with positive-confidence. NPU, which uses positive-confidence in the noisy PU learning framework, also did not work well since it is not designed for either AUC maximization or the SAR setting. The proposed method often outperformed w/oConf, which is the proposed method without positive-confidence. Thus, this result



(a) Mnist (b) Cifar10 (c) Diabetes (d) Blood

Figure 2: The average test AUCs and their standard errors over different positive class-priors for different parameters k of the transformation $h(r) = r^k$. Due to space limitation, we reported the results on two image and two tabular datasets; results on all datasets are provided in Section F.7.

Table 2: Results on the Cifar10-H and Fmnist-H datasets: average test AUCs over different positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$. We set $c = p(o = 1|y = 1) = 0.1$. Values in bold are not statistically different at the 5% level from the best performing method in each row according to a paired t-test.

Data	Ours	w/oConf	NTC	nnPU	PUAUC	PUSB	PG	Pconf	NPU
Cifar10-H	**0.5991**	**0.5998**	0.4779	0.4404	0.5813	0.5106	0.4779	0.5828	0.5712
Fmnist-H	**0.8285**	0.8192	0.7823	0.6013	**0.8324**	0.8242	0.7823	0.5211	0.8061

indicates the effectiveness of using positive-confidence information in our framework. Overall, these results show the effectiveness of the proposed method. The results with each π value are described in Section F.1. The proposed method worked well in each case.

We evaluated the performance of the proposed method when the confidence was distorted by strictly increasing transformation h. As described in Section 4.3, minimizing the proposed AUC risk with such transformed confidence still leads to the Bayes-optimal ranking for the AUC. Thus, the proposed method is expected to be empirically robust to such transformations. We considered transformation $h(r) = r^k$, where $k > 0$, with $k = 1$ corresponding to the original confidence. Figure 2 shows the average test AUCs with their standard errors when changing k of the transformation $h(r) = r^k$. Overall, the performance of the proposed method remained relatively stable across different transformations and showed no systematic degradation as k moved away from 1. In contrast, the performances of Pconf that also uses positive-confidence substantially degraded for some values of k because it does not have the ranking-invariance property of the proposed method. Results for other transformations are also reported in Section F.7. We also evaluated the robustness of the proposed method to additive Gaussian noise in positive-confidence in Section F.8.

In our experiments so far, we have used classifier-based confidence. However, confidence can also be obtained from labels provided by multiple annotators. Therefore, we evaluated this setting using the Cifar10-H and Fmnist-H datasets, where each instance was labeled by multiple real-world human annotators (Peterson et al., 2019; Ishida et al., 2023), and which have been widely used in confidence-based learning studies (Ishida et al., 2023; Ushio et al., 2026). Confidence for each instance can be obtained by averaging the corresponding annotations (Ishida et al., 2023). Details of the datasets and data construction are provided in Section E.2. Table 2 shows the results. On both datasets, the proposed method performed well, demonstrating its applicability to human-derived confidence. On Cifar10-H, the average positive-confidence value was close to one. Therefore, the proposed method behaved similarly to w/oConf, which sets all positive-confidence values to one, resulting in similar performance.

6 Conclusion

In this paper, we proposed an AUC maximization method under the SAR setting with biased PU data and positive-confidence. By using positive-confidence, the proposed method can maximize the AUC even in the SAR setting without restrictive assumptions about the data distributions or labeling mechanism. Our experiments on eight real-world datasets demonstrate the effectiveness of the proposed method.

AI USE STATEMENT

In this work, we used generative AI tools in the theoretical analysis section to assist in developing some theoretical derivations and writing proofs. We also used generative AI tools to edit the paper to improve readability and clarity. We carefully verified and, where necessary, corrected the proofs. The problem formulation, proposed risk rewriting, ranking-invariance property, and methodological ideas were developed by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve any new collection of personal data or human-subject experiments. All experiments use existing benchmark datasets, including datasets with previously collected human annotations. The proposed method is a general machine learning approach and does not introduce application-specific ethical risks beyond those of existing PU learning methods. However, when applied to sensitive domains such as medical care, cyber security, or fraud detection, biases in the observed labels or confidence information may be inherited by the learned model, and appropriate domain-specific validation and safeguards should be considered.

REPRODUCIBILITY STATEMENT

For reproducibility, we described details of the datasets in Sections 5.1, E.1, and E.2, and details of the neural network architectures and training procedures in Section E.3. In addition, we provided the pseudo-code for the training procedure in Algorithm 1. The assumptions used in the proposed method are described in Section 4, and theoretical analyses and their proofs are provided in Section B.

REFERENCES

Sikha Bagui and Kunqi Li. Resampling imbalanced data for network intrusion detection datasets. *Journal of Big Data*, 8(1):6, 2021.

Jessa Bekker and Jesse Davis. Learning from positive and unlabeled data: A survey. *Machine Learning*, 109(4):719–760, 2020.

Jessa Bekker, Pieter Robberechts, and Jesse Davis. Beyond the selected completely at random assumption for learning from positive and unlabeled data. In *ECML*, 2019.

Antonin Berthon, Bo Han, Gang Niu, Tongliang Liu, and Masashi Sugiyama. Confidence scores make instance-dependent label-noise learning possible. In *ICML*, 2021.

Andrew P Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159, 1997.

Ulf Brefeld, Tobias Scheffer, et al. Auc maximizing support vector learning. In *Proceedings of the ICML 2005 workshop on ROC Analysis in Machine Learning*, 2005.

Yuzhou Cao, Lei Feng, Yitian Xu, Bo An, Gang Niu, and Masashi Sugiyama. Learning from similarity-confidence data. In *ICML*, 2021.

Nontawat Charoenphakdee, Jongyeong Lee, and Masashi Sugiyama. On symmetric losses for learning from corrupted labels. In *ICML*, 2019.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

Marthinus Du Plessis, Gang Niu, and Masashi Sugiyama. Convex formulation for learning from positive and unlabeled data. In *ICML*, 2015.

Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *SIGKDD*, 2008.

Lei Feng, Senlin Shu, Nan Lu, Bo Han, Miao Xu, Gang Niu, Bo An, and Masashi Sugiyama. Pointwise binary classification with pairwise confidence comparisons. In *ICML*, 2021.

Akinori Fujino and Naonori Ueda. A semi-supervised auc optimization method with generative models. In *ICDM*, 2016.

Josh Gardner, Zoran Popovic, and Ludwig Schmidt. Benchmarking distribution shift in tabular data with tableshift. *NeurIPS*, 2023.

Walter Gerych, Thomas Hartvigsen, Luke Buquicchio, Emmanuel Agu, and Elke Rundensteiner. Recovering the propensity score from biased positive unlabeled data. In *AAAI*, 2022.

Chen Gong, Qizhou Wang, Tongliang Liu, Bo Han, Jane You, Jian Yang, and Dacheng Tao. Instance-dependent positive and unlabeled learning with labeling bias estimation. *IEEE transactions on pattern analysis and machine intelligence*, 44(8):4163–4177, 2021.

Fengxiang He, Tongliang Liu, Geoffrey I Webb, and Dacheng Tao. Instance-dependent pu learning by bayesian optimal relabeling. *arXiv preprint arXiv:1808.02180*, 2018.

Takashi Ishida, Gang Niu, and Masashi Sugiyama. Binary classification from positive-confidence data. In *NeurIPS*, 2018.

Takashi Ishida, Ikko Yamane, Nontawat Charoenphakdee, Gang Niu, and Masashi Sugiyama. Is the performance of my deep network too good to be true? a direct approach to estimating the bayes error in binary classification. In *ICLR*, 2023.

Fatemeh Jalalvand, Mohan Baruwal Chhetri, Surya Nepal, and Cecile Paris. Alert prioritisation in security operations centres: a systematic survey on criteria and methods. *ACM Computing Surveys*, 57(2):1–36, 2024.

Yangbangyan Jiang, Qianqian Xu, Yunrui Zhao, Zhiyong Yang, Peisong Wen, Xiaochun Cao, and Qingming Huang. Positive-unlabeled learning with label distribution alignment. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Justin M Johnson and Taghi M Khoshgoftaar. Survey on deep learning with class imbalance. *Journal of Big Data*, 6(1):1–54, 2019.

Takafumi Kanamori, Shohei Hido, and Masashi Sugiyama. A least-squares approach to direct importance estimation. *The Journal of Machine Learning Research*, 10:1391–1445, 2009.

Alex Kantchelian, Michael Carl Tschantz, Sadia Afroz, Brad Miller, Vaishaal Shankar, Rekha Bachwani, Anthony D Joseph, and J Doug Tygar. Better malware ground truth: Techniques for weighting anti-virus vendor labels. In *AISec*, 2015.

Masahiro Kato and Takeshi Teshima. Non-negative bregman divergence minimization for deep direct density ratio estimation. In *ICML*, 2021.

Masahiro Kato, Takeshi Teshima, and Junya Honda. Learning from positive and unlabeled data with a selection bias. In *ICLR*, 2019.

Diederik P Kingma and Jimmy Ba. Adam: a method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Ryuichi Kiryo, Gang Niu, Marthinus C Du Plessis, and Masashi Sugiyama. Positive-unlabeled learning with non-negative risk estimator. *NeurIPS*, 2017.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Atsutoshi Kumagai, Tomoharu Iwata, Hiroshi Takahashi, Taishi Nishiyama, and Yasuhiro Fujiwara. Auc maximization under positive distribution shift. In *NeurIPS*, 2024.

Atsutoshi Kumagai, Tomoharu Iwata, Hiroshi Takahashi, Taishi Nishiyama, Kazuki Adachi, and Yasuhiro Fujiwara. Positive-unlabeled auc maximization under covariate shift. In *ICML*, 2025a.

Atsutoshi Kumagai, Tomoharu Iwata, Hiroshi Takahashi, Taishi Nishiyama, and Yasuhiro Fujiwara. Importance-weighted positive-unlabeled learning for distribution shift adaptation. In *AISTATS*, 2025b.

Khiem H Le, Tuan V Tran, Hieu H Pham, Hieu T Nguyen, Tung T Le, and Ha Q Nguyen. Learning from multiple expert annotators for enhancing anomaly detection in medical image analysis. *IEEE Access*, 11:14105–14114, 2023.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017.

Mingrui Liu, Zhuoning Yuan, Yiming Ying, and Tianbao Yang. Stochastic auc maximization with deep neural networks. In *ICLR*, 2020.

Matthew McDermott, Lasse Hyldig Hansen, Haoran Zhang, Giovanni Angelotti, and Jack Gallifant. A closer look at auroc and auprc under class imbalance. In *NeurIPS*, 2024.

Giovanna Menardi and Nicola Torelli. Training and assessing classification rules with imbalanced data. *Data mining and knowledge discovery*, 28:92–122, 2014.

Aditya Krishna Menon and Robert C Williamson. Bipartite ranking: a risk-theoretic perspective. *Journal of Machine Learning Research*, 17(195):1–102, 2016.

Yisroel Mirsky, Tomer Doitshman, Yuval Elovici, and Asaf Shabtai. Kitsune: an ensemble of autoencoders for online network intrusion detection. In *NDSS*, 2018.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, pp. 7. Granada, Spain, 2011.

Je-Kang Park, Bae-Keun Kwon, Jun-Hyub Park, and Dong-Joong Kang. Machine learning-based imaging system for surface defect inspection. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 3:303–310, 2016.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Joshua C Peterson, Ruairidh M Battleday, Thomas L Griffiths, and Olga Russakovsky. Human uncertainty makes classification more robust. In *ICCV*, 2019.

Tomoya Sakai and Nobuyuki Shimizu. Covariate shift adaptation on learning from positive and unlabeled data. In *AAAI*, 2019.

Tomoya Sakai, Gang Niu, and Masashi Sugiyama. Semi-supervised auc optimization based on positive-unlabeled learning. *Machine Learning*, 107:767–794, 2018.

Ryoma Sato. Interestingness first classifiers. *arXiv preprint arXiv:2508.19780*, 2025.

Kazuhiko Shinoda, Hirotaka Kaji, and Masashi Sugiyama. Binary classification from positive data with skewed confidence. In *IJCAI*, 2021.

Guangxin Su, Weitong Chen, and Miao Xu. Positive-unlabeled learning from imbalanced data. In *IJCAI*, 2021.

Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

Masashi Sugiyama, Han Bao, Takashi Ishida, Nan Lu, and Tomoya Sakai. *Machine learning from weak supervision: an empirical risk minimization approach*. MIT Press, 2022.

Xijia Tang, Chao Xu, Hong Tao, Xiaoyu Ma, and Chenping Hou. Confidence-based pu learning with instance-dependent label noise. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

Paweł Teisseyre, Timo Martens, Jessa Bekker, and Jesse Davis. Learning from biased positive-unlabeled data via threshold calibration. In *AISTATS*, 2025.

Ryota Ushio, Takashi Ishida, and Masashi Sugiyama. Practical estimation of the optimal classification error with soft labels and calibration. In *ICLR*, 2026.

Guanjin Wang, Stephen Wai Hang Kwok, Mohammed Yousufuddin, and Ferdous Sohel. A novel auc maximization imbalanced learning approach for predicting composite outcomes in covid-19 hospitalized patients. *IEEE journal of biomedical and health informatics*, 2023a.

Wei Wang, Lei Feng, Yuchen Jiang, Gang Niu, Min-Ling Zhang, and Masashi Sugiyama. Binary classification with confidence difference. In *NeurIPS*, 2023b.

Gill Ward, Trevor Hastie, Simon Barry, Jane Elith, and John R Leathwick. Presence-only data and the em algorithm. *Biometrics*, 65(2):554–563, 2009.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Zheng Xie and Ming Li. Semi-supervised auc optimization without guessing labels of unlabeled data. In *AAAI*, 2018.

Zheng Xie, Yu Liu, Hao-Yuan He, Ming Li, and Zhi-Hua Zhou. Weakly supervised auc optimization: a unified partial auc approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Chen Yang, Minghan Zhao, Chenyu Zhu, Suiwei Xie, and Yifei Chen. Automated detection of breast cancer metastases. In *DSInS*, 2021.

Tianbao Yang and Yiming Ying. Auc maximization in the era of big data and ai: A survey. *ACM computing surveys*, 55(8):1–37, 2022.

Yiming Ying, Longyin Wen, and Siwei Lyu. Stochastic online auc maximization. *NeurIPS*, 2016.

Zhuoning Yuan, Zhishuai Guo, Nitesh Chawla, and Tianbao Yang. Compositional training for end-to-end deep auc maximization. In *ICLR*, 2021a.

Zhuoning Yuan, Yan Yan, Milan Sonka, and Tianbao Yang. Large-scale robust deep auc maximization: a new surrogate loss and empirical studies on medical image classification. In *ICCV*, 2021b.

A Extended Related Work

Kumagai et al. (2025a) have recently proposed a distribution shift adaptation method for AUC maximization. Specifically, as distribution shift, this paper assumes the covariate shift, which refers to the situation in which the distribution of input instance changes while the input-output (class label) relationship remains unchanged between the training and test phases. This method uses not only PU data from the training phase but also unlabeled data from the test phase to address the covariate shift. The PU data in the training phase can be regarded as biased data. Specifically, although the distributions of labeled positive data and unlabeled positive data in the training phase are identical (i.e., the SCAR setting), the distributions of labeled positive data in the training phase and the unlabeled positive data in the test phase may differ due to the covariate shift (i.e., a SAR-like setting). While this approach requires a large amount of unlabeled data obtained in the training phase to handle covariate shift, such unlabeled data are not available in our problem setting. Instead, the proposed method requires confidence information for a small amount of positive data.

B THEORETICAL ANALYSIS

We first show a basic relationship between the supports of the labeling probability and the posterior probability, and derive an expectation identity used in the following analysis.

Lemma B.1. *Let $\eta(\mathbf{x}) := p(y = 1|\mathbf{x})$, $u(\mathbf{x}) := p(o = 1|\mathbf{x})$, and $\alpha := p(o = 1)$. Under the PU assumption and the positivity condition in Section 4.1,*

$$I(u(\mathbf{x}) > 0) = I(\eta(\mathbf{x}) > 0) \quad p\text{-almost surely}, \tag{19}$$

and $u(\mathbf{x}) > 0$ for p^{l}-almost every $\mathbf{x}$. Moreover, for any measurable function φ such that the following expectations are finite,

$$\mathbb{E}_{\mathbf{x}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\frac{\varphi(\mathbf{x})}{u(\mathbf{x})}\right] = \frac{1}{\alpha}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\varphi(\mathbf{x})I(\eta(\mathbf{x}) > 0)\right]. \tag{20}$$

Proof. By Eq. (96) in Section C.3,

$$u(\mathbf{x}) = e(\mathbf{x})\eta(\mathbf{x}). \tag{21}$$

Therefore, $u(\mathbf{x}) > 0$ implies $\eta(\mathbf{x}) > 0$.

To show the reverse implication, the positivity condition implies

$$\int_{\{\mathbf{x}:e(\mathbf{x})=0\}} p^{\mathrm{p}}(\mathbf{x})d\mathbf{x} = 0. \tag{22}$$

Therefore, letting $A := \{\mathbf{x} : e(\mathbf{x}) = 0,\ \eta(\mathbf{x}) > 0\}$, we have

$$\begin{aligned} 0 &= \int_A p^{\mathrm{p}}(\mathbf{x})d\mathbf{x} \\ &= \frac{1}{\pi}\int_A \eta(\mathbf{x})p(\mathbf{x})d\mathbf{x}, \end{aligned} \tag{23}$$

where we used $p^{\mathrm{p}}(\mathbf{x}) = \eta(\mathbf{x})p(\mathbf{x})/\pi$. Since $\eta(\mathbf{x}) > 0$ on A, this implies

$$\int_A p(\mathbf{x})d\mathbf{x} = 0. \tag{24}$$

Thus, $e(\mathbf{x}) > 0$ for p-almost every $\mathbf{x}$ such that $\eta(\mathbf{x}) > 0$. Since $u(\mathbf{x}) = e(\mathbf{x})\eta(\mathbf{x})$, it follows that

$$\eta(\mathbf{x}) > 0 \quad \Longrightarrow \quad u(\mathbf{x}) > 0 \qquad p\text{-almost surely}. \tag{25}$$

Combining the two implications, we obtain

$$I(u(\mathbf{x}) > 0) = I(\eta(\mathbf{x}) > 0) \quad p\text{-almost surely}. \tag{26}$$

Next, by Bayes' rule,

$$p^{\mathrm{l}}(\mathbf{x}) = \frac{u(\mathbf{x})p(\mathbf{x})}{\alpha}. \tag{27}$$

Therefore,

$$\begin{aligned} \int_{\{\mathbf{x}:u(\mathbf{x})=0\}} p^{\mathrm{l}}(\mathbf{x})d\mathbf{x} &= \frac{1}{\alpha}\int_{\{\mathbf{x}:u(\mathbf{x})=0\}} u(\mathbf{x})p(\mathbf{x})d\mathbf{x} \\ &= 0. \end{aligned} \tag{28}$$

Thus, $u(\mathbf{x}) > 0$ for p^{l}-almost every $\mathbf{x}$. We can therefore restrict the expectation under p^{l} to the region where $u(\mathbf{x}) > 0$. Using Eq. (27), we obtain

$$\begin{aligned} &\mathbb{E}_{\mathbf{x}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\frac{\varphi(\mathbf{x})}{u(\mathbf{x})}\right] \\ &= \int_{\{\mathbf{x}:u(\mathbf{x})>0\}} p^{\mathrm{l}}(\mathbf{x})\frac{\varphi(\mathbf{x})}{u(\mathbf{x})}d\mathbf{x} \\ &= \frac{1}{\alpha}\int_{\{\mathbf{x}:u(\mathbf{x})>0\}} p(\mathbf{x})\varphi(\mathbf{x})d\mathbf{x} \\ &= \frac{1}{\alpha}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\varphi(\mathbf{x})I(u(\mathbf{x}) > 0)\right] \\ &= \frac{1}{\alpha}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\varphi(\mathbf{x})I(\eta(\mathbf{x}) > 0)\right], \end{aligned} \tag{29}$$

where the last equality follows from Eq. (19). □

B.1 Proof of The Robustness Statement under Strictly Increasing Transformations of Positive-confidence

Proof. Let $\eta(\mathbf{x}) := p(y = 1|\mathbf{x})$ and $u(\mathbf{x}) := p(o = 1|\mathbf{x})$. Since the multiplicative coefficient in Eq. (18) is positive and the additive term is independent of s, minimizing $\mathcal{R}^h_\sigma(s)$ is equivalent to minimizing

$$\widetilde{\mathcal{R}}^h_\sigma(s) := \mathbb{E}_{\mathbf{x}^\mathrm{p}\sim p^1(\mathbf{x})}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\frac{h(\eta(\mathbf{x}^\mathrm{p}))}{u(\mathbf{x}^\mathrm{p})}f(\mathbf{x}^\mathrm{p},\mathbf{x})\right]. \tag{30}$$

Applying Lemma B.1 with

$$\varphi(\mathbf{x}^\mathrm{p}) = h(\eta(\mathbf{x}^\mathrm{p}))\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[f(\mathbf{x}^\mathrm{p},\mathbf{x})], \tag{31}$$

we obtain

$$\widetilde{\mathcal{R}}^h_\sigma(s) = \frac{1}{\alpha}\mathbb{E}_{\mathbf{x}^\mathrm{p},\mathbf{x}\sim p(\mathbf{x})}\left[h(\eta(\mathbf{x}^\mathrm{p}))I(\eta(\mathbf{x}^\mathrm{p}) > 0)f(\mathbf{x}^\mathrm{p},\mathbf{x})\right]. \tag{32}$$

Define

$$g_h(z) := h(z)I(z > 0), \tag{33}$$

and

$$Z_h := \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[g_h(\eta(\mathbf{x}))], \qquad q_h(\mathbf{x}) := \frac{g_h(\eta(\mathbf{x}))p(\mathbf{x})}{Z_h}. \tag{34}$$

Since $\pi = \mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\eta(\mathbf{x})] > 0$ and $g_h(z) > 0$ for $z > 0$, we have $Z_h > 0$. Then,

$$\widetilde{\mathcal{R}}^h_\sigma(s) = \frac{Z_h}{\alpha}\mathbb{E}_{\mathbf{x}^\mathrm{p}\sim q_h(\mathbf{x})}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[f(\mathbf{x}^\mathrm{p},\mathbf{x})]. \tag{35}$$

Thus, up to a positive multiplicative constant, $\widetilde{\mathcal{R}}^h_\sigma$ is the sigmoid AUC surrogate for bipartite ranking between q_h and p. By the AUC consistency of the sigmoid loss (Charoenphakdee et al., 2019) and the Bayes characterization of bipartite ranking (Menon & Williamson, 2016), the Bayes-optimal ranking for this problem is determined by the ordering of

$$\frac{q_h(\mathbf{x})}{p(\mathbf{x})} = \frac{g_h(\eta(\mathbf{x}))}{Z_h}. \tag{36}$$

Since $h : [0,1] \to [0,1]$ is strictly increasing, g_h is also strictly increasing on $[0,1]$: for $z > 0$, $g_h(z) = h(z) > h(0) \geq 0 = g_h(0)$, and for $0 < z_1 < z_2$, $g_h(z_1) = h(z_1) < h(z_2) = g_h(z_2)$. Thus, $g_h(\eta(\mathbf{x}))$ induces the same ordering as $\eta(\mathbf{x})$. Since $Z_h > 0$, the Bayes-optimal ranking induced by $q_h(\mathbf{x})/p(\mathbf{x})$ is identical to that induced by $\eta(\mathbf{x})$.

On the other hand, the Bayes-optimal ranking for the original AUC risk in Eq. (9) is also determined by the ordering of $\eta(\mathbf{x})$ (Menon & Williamson, 2016). Therefore, the transformed objective and the original AUC risk induce the same Bayes-optimal ranking for the AUC. □

B.2 Sensitivity to Labeling-Probability Estimation

In practice, the labeling probability $p(o = 1|\mathbf{x})$ in Eq. (15) is unknown and must be estimated from the given PU data. Here, we analyze how its estimation error affects the rewritten AUC risk and the resulting ranking.

Proposition B.2. *Let $\eta(\mathbf{x}) := p(y = 1|\mathbf{x})$ and $u(\mathbf{x}) := p(o = 1|\mathbf{x})$. For an estimate $\widehat{u}(\mathbf{x})$ of $u(\mathbf{x})$, define*

$$\mathcal{R}_\sigma(s;\widehat{u}) := \frac{c}{1-\pi}\mathbb{E}_{\mathbf{x}^\mathrm{p}\sim p^1(\mathbf{x})}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\frac{\eta(\mathbf{x}^\mathrm{p})}{\widehat{u}(\mathbf{x}^\mathrm{p})}f(\mathbf{x}^\mathrm{p},\mathbf{x})\right] - \frac{\pi}{2(1-\pi)}. \tag{37}$$

Note that $\mathcal{R}_\sigma(s;u) = \mathcal{R}_\sigma(s)$ in Eq. (15)*. Suppose that $\widehat{u}(\mathbf{x}) \geq \delta > 0$ for all $\mathbf{x}$ and*

$$|\widehat{u}(\mathbf{x}) - u(\mathbf{x})| \leq \epsilon \tag{38}$$

for p-almost every $\mathbf{x}$ satisfying $u(\mathbf{x}) > 0$. Then,

$$\sup_s |\mathcal{R}_\sigma(s;\widehat{u}) - \mathcal{R}_\sigma(s;u)| \leq \frac{\epsilon}{(1-\pi)\delta}. \tag{39}$$

Furthermore, suppose that $\widehat{s}$ is an approximate minimizer satisfying

$$\mathcal{R}_\sigma(\widehat{s};\widehat{u}) \leq \inf_s \mathcal{R}_\sigma(s;\widehat{u}) + \xi_{\mathrm{opt}}, \tag{40}$$

where $\xi_{\mathrm{opt}} \geq 0$. Then,

$$\mathcal{R}_\sigma(\widehat{s};u) - \inf_s \mathcal{R}_\sigma(s;u) \leq \frac{2\epsilon}{(1-\pi)\delta} + \xi_{\mathrm{opt}}. \tag{41}$$

Proof. Since $0 \leq f \leq 1$,

$$\begin{aligned} &|\mathcal{R}_\sigma(s;\widehat{u}) - \mathcal{R}_\sigma(s;u)| \\ &\leq \frac{c}{1-\pi}\mathbb{E}_{\mathbf{x}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\eta(\mathbf{x})\frac{|\widehat{u}(\mathbf{x}) - u(\mathbf{x})|}{u(\mathbf{x})\widehat{u}(\mathbf{x})}\right]. \end{aligned} \tag{42}$$

Applying Lemma B.1 with

$$\varphi(\mathbf{x}) = \eta(\mathbf{x})\frac{|\widehat{u}(\mathbf{x}) - u(\mathbf{x})|}{\widehat{u}(\mathbf{x})}, \tag{43}$$

we obtain

$$\begin{aligned} &|\mathcal{R}_\sigma(s;\widehat{u}) - \mathcal{R}_\sigma(s;u)| \\ &\leq \frac{c}{\alpha(1-\pi)}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\eta(\mathbf{x})\frac{|\widehat{u}(\mathbf{x}) - u(\mathbf{x})|}{\widehat{u}(\mathbf{x})}I(\eta(\mathbf{x}) > 0)\right] \\ &\leq \frac{c\epsilon}{\alpha(1-\pi)\delta}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\eta(\mathbf{x})I(\eta(\mathbf{x}) > 0)\right] \\ &= \frac{c\epsilon}{\alpha(1-\pi)\delta}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\eta(\mathbf{x})] = \frac{\epsilon}{(1-\pi)\delta}, \end{aligned} \tag{44}$$

where the second inequality follows from $I(\eta(\mathbf{x}) > 0) = I(u(\mathbf{x}) > 0)$ p-almost surely in Lemma B.1 and the assumptions on $\widehat{u}$. In the last equality, we used $\mathbb{E}_p[\eta(\mathbf{x})] = \pi$ and $\alpha = c\pi$. By taking the supremum over s, we obtain Eq. (39).

By Eq. (39),

$$\begin{aligned} \mathcal{R}_\sigma(\widehat{s};u) &\leq \mathcal{R}_\sigma(\widehat{s};\widehat{u}) + \frac{\epsilon}{(1-\pi)\delta} \\ &\leq \inf_s \mathcal{R}_\sigma(s;\widehat{u}) + \xi_{\mathrm{opt}} + \frac{\epsilon}{(1-\pi)\delta} \\ &\leq \inf_s \mathcal{R}_\sigma(s;u) + \xi_{\mathrm{opt}} + \frac{2\epsilon}{(1-\pi)\delta}, \end{aligned} \tag{45}$$

which proves the second claim. □

Eq. (39) shows that, for fixed π and δ, the deviation of the rewritten AUC risk increases at most linearly with the labeling probability estimation error ϵ. The factor $1/\delta$ also indicates that the effect of estimation errors can be amplified when the estimated labeling probability is close to zero. This provides theoretical motivation for the clipping used in our experiments.

We next separately analyze how misspecification of the labeling probability affects the resulting ranking. Using $p^{\mathrm{l}}(\mathbf{x}) = u(\mathbf{x})p(\mathbf{x})/\alpha$, the part of $\mathcal{R}_\sigma(s;\widehat{u})$ that depends on s can, up to a positive multiplicative constant, be written as

$$\mathbb{E}_{\mathbf{x}^{\mathrm{p}},\mathbf{x}\sim p(\mathbf{x})}\left[\eta(\mathbf{x}^{\mathrm{p}})\frac{u(\mathbf{x}^{\mathrm{p}})}{\widehat{u}(\mathbf{x}^{\mathrm{p}})}f(\mathbf{x}^{\mathrm{p}},\mathbf{x})\right]. \tag{46}$$

Define

$$\widetilde{\eta}(\mathbf{x}) := \eta(\mathbf{x})\frac{u(\mathbf{x})}{\widehat{u}(\mathbf{x})}. \tag{47}$$

Using the same bipartite-ranking argument as in Section B.1, the Bayes-optimal ranking associated with this objective is determined by the ordering of $\widetilde{\eta}(\mathbf{x})$.

This result shows that some estimation errors in the labeling probability do not affect the resulting ranking. First, suppose that $\widehat{u}(\mathbf{x}) = au(\mathbf{x})$ for some constant $a > 0$ for p-almost every $\mathbf{x}$ satisfying $u(\mathbf{x}) > 0$. On this set,

$$\widetilde{\eta}(\mathbf{x}) = \eta(\mathbf{x})\frac{u(\mathbf{x})}{\widehat{u}(\mathbf{x})} = \frac{\eta(\mathbf{x})}{a}. \tag{48}$$

On the other hand, by Lemma B.1, $u(\mathbf{x}) = 0$ implies $\eta(\mathbf{x}) = 0$ p-almost surely. In addition, since $\widetilde{\eta}(\mathbf{x}) = 0$ when $u(\mathbf{x}) = 0$, we have

$$\widetilde{\eta}(\mathbf{x}) = \frac{\eta(\mathbf{x})}{a} \quad p\text{-almost surely.} \tag{49}$$

Since the Bayes-optimal ranking is unaffected by changes on a p-null set, the above equality and $a > 0$ imply that $\widetilde{\eta}$ induces the same ordering as η. Thus, a constant multiplicative error in the estimated labeling probability does not change the Bayes-optimal ranking for the AUC.

More generally, suppose that the relative estimation error is uniformly bounded as

$$\left|\frac{\widehat{u}(\mathbf{x})}{u(\mathbf{x})} - 1\right| \le \rho < 1 \tag{50}$$

for p-almost every $\mathbf{x}$ satisfying $u(\mathbf{x}) > 0$. Then,

$$1 - \rho \le \frac{\widehat{u}(\mathbf{x})}{u(\mathbf{x})} \le 1 + \rho, \tag{51}$$

and thus

$$\frac{\eta(\mathbf{x})}{1+\rho} \le \widetilde{\eta}(\mathbf{x}) \le \frac{\eta(\mathbf{x})}{1-\rho}. \tag{52}$$

For two instances $\mathbf{x}$ and $\bar{\mathbf{x}}$ outside the corresponding p-null set satisfying $\eta(\mathbf{x}) > \eta(\bar{\mathbf{x}}) > 0$, their ordering is guaranteed to be preserved if the smallest possible value of $\widetilde{\eta}(\mathbf{x})$ is larger than the largest possible value of $\widetilde{\eta}(\bar{\mathbf{x}})$, namely, if

$$\frac{\eta(\mathbf{x})}{1+\rho} > \frac{\eta(\bar{\mathbf{x}})}{1-\rho}, \tag{53}$$

or equivalently,

$$\frac{\eta(\mathbf{x})}{\eta(\bar{\mathbf{x}})} > \frac{1+\rho}{1-\rho}. \tag{54}$$

If $\eta(\bar{\mathbf{x}}) = 0 < \eta(\mathbf{x})$, their ordering is also preserved outside the corresponding p-null set. Therefore, under the relative-error bound in Eq. (50), a pair can have its ordering reversed only when their posterior probabilities are sufficiently close.

B.3 Generalization Error Analysis

We next provide a finite-sample analysis of learning with the proposed AUC risk estimator. We consider finite score function and labeling probability classes to analyze the basic convergence behavior of the estimator.

Let $\mathcal{S}$ be a finite class of score functions, and let $\mathcal{U}_\delta$ be a finite class of labeling probability functions such that

$$v(\mathbf{x}) \ge \delta > 0 \quad \text{for all } v \in \mathcal{U}_\delta \text{ and } \mathbf{x}. \tag{55}$$

Define $K := |\mathcal{S}||\mathcal{U}_\delta|$. Let $\widetilde{r}(\mathbf{x}) \in [0,1]$ denote the observed, possibly noisy confidence. To make the dependence on the score function explicit, we write

$$f_s(\mathbf{x}, \mathbf{x}') := \sigma(-s(\mathbf{x}) + s(\mathbf{x}')). \tag{56}$$

For $s \in \mathcal{S}$ and $v \in \mathcal{U}_\delta$, define

$$h_{s,v}(\mathbf{x}, \mathbf{x}') := \frac{\widetilde{r}(\mathbf{x})}{v(\mathbf{x})} f_s(\mathbf{x}, \mathbf{x}'). \tag{57}$$

Since $0 \le \widetilde{r}(\mathbf{x}) \le 1$, $0 \le f_s(\mathbf{x}, \mathbf{x}') \le 1$, and $v(\mathbf{x}) \ge \delta$, we have

$$0 \le h_{s,v}(\mathbf{x}, \mathbf{x}') \le \frac{1}{\delta}. \tag{58}$$

Define the empirical positive-positive and positive-unlabeled terms corresponding to Eq. (17) as

$$\widehat{L}_{\mathrm{pp}}(s,v) := \frac{1}{N^{\mathrm{p}}(N^{\mathrm{p}}-1)} \sum_{n \ne m}^{N^{\mathrm{p}}, N^{\mathrm{p}}} h_{s,v}(\mathbf{x}^{\mathrm{p}}_n, \mathbf{x}^{\mathrm{p}}_m), \tag{59}$$

$$\widehat{L}_{\mathrm{pu}}(s,v) := \frac{1}{N^{\mathrm{p}} N} \sum_{n=1}^{N^{\mathrm{p}}} \sum_{m=1}^{N} h_{s,v}(\mathbf{x}^{\mathrm{p}}_n, \mathbf{x}_m). \tag{60}$$

The corresponding population terms are

$$L_{\mathrm{pp}}(s,v) := \mathbb{E}_{\mathbf{x} \sim p^{\mathrm{l}}(\mathbf{x})} \mathbb{E}_{\mathbf{x}' \sim p^{\mathrm{l}}(\mathbf{x})} [h_{s,v}(\mathbf{x}, \mathbf{x}')], \tag{61}$$

$$L_{\mathrm{pu}}(s,v) := \mathbb{E}_{\mathbf{x} \sim p^{\mathrm{l}}(\mathbf{x})} \mathbb{E}_{\mathbf{x}' \sim p^{\mathrm{u}}(\mathbf{x})} [h_{s,v}(\mathbf{x}, \mathbf{x}')]. \tag{62}$$

Let $\alpha := p(o = 1)$ and define

$$L(s,v) := \alpha L_{\mathrm{pp}}(s,v) + (1-\alpha) L_{\mathrm{pu}}(s,v). \tag{63}$$

We first derive a uniform finite-sample bound. For $\tau \in (0,1)$, define

$$B_{N^{\mathrm{p}},N}(\tau) := \frac{1}{\delta} \left[\alpha \sqrt{\frac{2\log(4K/\tau)}{N^{\mathrm{p}}}} + (1-\alpha) \sqrt{\frac{\log(4K/\tau)}{2} \left(\frac{1}{N^{\mathrm{p}}} + \frac{1}{N} \right)} \right]. \tag{64}$$

Lemma B.3. *With probability at least* $1-\tau$*,*

$$\sup_{\substack{s \in \mathcal{S} \\ v \in \mathcal{U}_\delta}} \left| \alpha \widehat{L}_{\mathrm{pp}}(s,v) + (1-\alpha) \widehat{L}_{\mathrm{pu}}(s,v) - L(s,v) \right| \le B_{N^{\mathrm{p}},N}(\tau). \tag{65}$$

Proof. Consider a fixed pair (s,v). Replacing one labeled positive instance changes at most $2(N^{\mathrm{p}}-1)$ summands in $\widehat{L}_{\mathrm{pp}}(s,v)$. Thus, by Eq. (58), replacing one labeled positive instance changes $\widehat{L}_{\mathrm{pp}}(s,v)$ by at most $2/(N^{\mathrm{p}}\delta)$. As a result, McDiarmid's inequality gives

$$\Pr\left(\left| \widehat{L}_{\mathrm{pp}}(s,v) - L_{\mathrm{pp}}(s,v) \right| \ge t \right) \le 2\exp\left(-\frac{N^{\mathrm{p}}\delta^2 t^2}{2} \right). \tag{66}$$

For $\widehat{L}_{\mathrm{pu}}(s,v)$, replacing one labeled positive instance changes N summands and thus changes the average by at most $1/(N^{\mathrm{p}}\delta)$. Similarly, replacing one unlabeled instance changes N^{p} summands and thus changes the average by at most $1/(N\delta)$. As a result, McDiarmid's inequality gives

$$\Pr\left(\left| \widehat{L}_{\mathrm{pu}}(s,v) - L_{\mathrm{pu}}(s,v) \right| \ge t \right) \le 2\exp\left(-\frac{2\delta^2 t^2}{1/N^{\mathrm{p}} + 1/N} \right). \tag{67}$$

Applying a union bound over the $K = |\mathcal{S}||\mathcal{U}_\delta|$ pairs in $\mathcal{S} \times \mathcal{U}_\delta$ and over the two empirical terms yields, with probability at least $1-\tau$,

$$\sup_{\substack{s \in \mathcal{S} \\ v \in \mathcal{U}_\delta}} \left| \widehat{L}_{\mathrm{pp}}(s,v) - L_{\mathrm{pp}}(s,v) \right| \le \frac{1}{\delta} \sqrt{\frac{2\log(4K/\tau)}{N^{\mathrm{p}}}}, \tag{68}$$

$$\sup_{\substack{s \in \mathcal{S} \\ v \in \mathcal{U}_\delta}} \left| \widehat{L}_{\mathrm{pu}}(s,v) - L_{\mathrm{pu}}(s,v) \right| \le \frac{1}{\delta} \sqrt{\frac{\log(4K/\tau)}{2} \left(\frac{1}{N^{\mathrm{p}}} + \frac{1}{N} \right)}. \tag{69}$$

Eq. (65) follows by combining these two inequalities with Eq. (63). □

In practice, α is estimated as

$$\widehat{\alpha} := \frac{N^{\mathrm{p}}}{N^{\mathrm{p}} + N}. \tag{70}$$

Define its estimation error as

$$\epsilon_\alpha := |\widehat{\alpha} - \alpha|. \tag{71}$$

Since $0 \le \widehat{L}_{\mathrm{pp}}(s, v), \widehat{L}_{\mathrm{pu}}(s, v) \le 1/\delta$, we have

$$\begin{aligned}
&\left|\widehat{\alpha}\widehat{L}_{\mathrm{pp}}(s, v) + (1 - \widehat{\alpha})\widehat{L}_{\mathrm{pu}}(s, v) - \alpha\widehat{L}_{\mathrm{pp}}(s, v) - (1 - \alpha)\widehat{L}_{\mathrm{pu}}(s, v)\right| \\
&= |\widehat{\alpha} - \alpha| \left|\widehat{L}_{\mathrm{pp}}(s, v) - \widehat{L}_{\mathrm{pu}}(s, v)\right| \\
&\le \frac{\epsilon_\alpha}{\delta}.
\end{aligned} \tag{72}$$

Thus, on the event in Eq. (65),

$$\begin{aligned}
&\sup_{\substack{s \in \mathcal{S} \\ v \in \mathcal{U}_\delta}} \left|\widehat{\alpha}\widehat{L}_{\mathrm{pp}}(s, v) + (1 - \widehat{\alpha})\widehat{L}_{\mathrm{pu}}(s, v) - L(s, v)\right| \\
&\qquad \le B_{N^{\mathrm{p}}, N}(\tau) + \frac{\epsilon_\alpha}{\delta}.
\end{aligned} \tag{73}$$

Thus, Eq. (73) also holds with probability at least $1 - \tau$.

For arbitrary confidence $\widetilde{r}$ and $v \in \mathcal{U}_\delta$, define the population and empirical rewritten risks by

$$\mathcal{R}_\sigma(s; \widetilde{r}, v) := \frac{c}{1 - \pi} L(s, v) - \frac{\pi}{2(1 - \pi)}, \tag{74}$$

$$\widehat{\mathcal{R}}_\sigma(s; \widetilde{r}, v) := \frac{c}{1 - \pi} \left[\widehat{\alpha}\widehat{L}_{\mathrm{pp}}(s, v) + (1 - \widehat{\alpha})\widehat{L}_{\mathrm{pu}}(s, v)\right] - \frac{\pi}{2(1 - \pi)}. \tag{75}$$

Define

$$\Gamma_{N^{\mathrm{p}}, N}(\tau) := \frac{c}{1 - \pi} \left[B_{N^{\mathrm{p}}, N}(\tau) + \frac{\epsilon_\alpha}{\delta}\right]. \tag{76}$$

Then, Eq. (73) implies that, with probability at least $1 - \tau$,

$$\sup_{\substack{s \in \mathcal{S} \\ v \in \mathcal{U}_\delta}} \left|\widehat{\mathcal{R}}_\sigma(s; \widetilde{r}, v) - \mathcal{R}_\sigma(s; \widetilde{r}, v)\right| \le \Gamma_{N^{\mathrm{p}}, N}(\tau). \tag{77}$$

We next define the population risk corresponding to the true confidence and labeling probability. By Lemma B.1, $u(\mathbf{x}) > 0$ for p^{l}-almost every $\mathbf{x}$, so the ratio $\eta(\mathbf{x})/u(\mathbf{x})$ in the following expectation is well-defined p^{l}-almost surely. We denote this population risk by

$$\mathcal{R}_\sigma(s; \eta, u) := \frac{c}{1 - \pi} \mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{l}}(\mathbf{x})} \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \left[\frac{\eta(\mathbf{x}^{\mathrm{p}})}{u(\mathbf{x}^{\mathrm{p}})} f_s(\mathbf{x}^{\mathrm{p}}, \mathbf{x})\right] - \frac{\pi}{2(1 - \pi)}. \tag{78}$$

By Eq. (15), $\mathcal{R}_\sigma(s; \eta, u)$ is equal to the original population AUC risk.

We next show the excess-risk bound for the score function learned with the proposed estimator.

Theorem B.4. *Suppose that the data-dependent estimate $\widehat{u} \in \mathcal{U}_\delta$ and that $\widehat{s} \in \mathcal{S}$ is an approximate empirical minimizer satisfying*

$$\widehat{\mathcal{R}}_\sigma(\widehat{s}; \widetilde{r}, \widehat{u}) \le \inf_{s \in \mathcal{S}} \widehat{\mathcal{R}}_\sigma(s; \widetilde{r}, \widehat{u}) + \xi_{\mathrm{opt}}, \tag{79}$$

where $\xi_{\mathrm{opt}} \ge 0$. Suppose that the PU assumption and the positivity condition in Section 4.1 hold, and that

$$\begin{aligned}
|\widetilde{r}(\mathbf{x}) - \eta(\mathbf{x})| &\le \epsilon_r, \\
|\widehat{u}(\mathbf{x}) - u(\mathbf{x})| &\le \epsilon_u
\end{aligned} \tag{80}$$

for p-almost every $\mathbf{x}$ satisfying $u(\mathbf{x}) > 0$. Then, with probability at least $1 - \tau$,

$$\mathcal{R}_\sigma(\widehat{s}; \eta, u) - \inf_{s \in \mathcal{S}} \mathcal{R}_\sigma(s; \eta, u) \le 2\Gamma_{N^{\mathrm{p}}, N}(\tau) + \frac{2(c\epsilon_r + \epsilon_u)}{(1 - \pi)\delta} + \xi_{\mathrm{opt}}. \tag{81}$$

Proof. On the event in Eq. (77), the uniform bound holds simultaneously for all $(s, v) \in \mathcal{S} \times \mathcal{U}_\delta$. Therefore, since $\widehat{u} \in \mathcal{U}_\delta$, it also holds for the data-dependent estimate $\widehat{u}$:

$$\sup_{s\in\mathcal{S}} \left|\widehat{\mathcal{R}}_\sigma(s;\widetilde{r},\widehat{u}) - \mathcal{R}_\sigma(s;\widetilde{r},\widehat{u})\right| \le \Gamma_{N^{\mathrm{P}},N}(\tau). \tag{82}$$

Since $\mathcal{S}$ is finite, let

$$s^\dagger \in \arg\min_{s\in\mathcal{S}} \mathcal{R}_\sigma(s;\widetilde{r},\widehat{u}). \tag{83}$$

Then, by Eqs. (82) and (79), we have

$$\begin{aligned}
\mathcal{R}_\sigma(\widehat{s};\widetilde{r},\widehat{u}) &\le \widehat{\mathcal{R}}_\sigma(\widehat{s};\widetilde{r},\widehat{u}) + \Gamma_{N^{\mathrm{P}},N}(\tau) \\
&\le \widehat{\mathcal{R}}_\sigma(s^\dagger;\widetilde{r},\widehat{u}) + \Gamma_{N^{\mathrm{P}},N}(\tau) + \xi_{\mathrm{opt}} \\
&\le \mathcal{R}_\sigma(s^\dagger;\widetilde{r},\widehat{u}) + 2\Gamma_{N^{\mathrm{P}},N}(\tau) + \xi_{\mathrm{opt}}.
\end{aligned} \tag{84}$$

By the definition of $s^\dagger$, this yields

$$\mathcal{R}_\sigma(\widehat{s};\widetilde{r},\widehat{u}) - \inf_{s\in\mathcal{S}} \mathcal{R}_\sigma(s;\widetilde{r},\widehat{u}) \le 2\Gamma_{N^{\mathrm{P}},N}(\tau) + \xi_{\mathrm{opt}}. \tag{85}$$

We next analyze the effects of the confidence error and the labeling-probability estimation error. For any $s \in \mathcal{S}$, using $0 \le f_s \le 1$, we have

$$\begin{aligned}
&|\mathcal{R}_\sigma(s;\widetilde{r},\widehat{u}) - \mathcal{R}_\sigma(s;\eta,u)| \\
&\le \frac{c}{1-\pi}\mathbb{E}_{\mathbf{x}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\left|\frac{\widetilde{r}(\mathbf{x})}{\widehat{u}(\mathbf{x})} - \frac{\eta(\mathbf{x})}{u(\mathbf{x})}\right|\right].
\end{aligned} \tag{86}$$

By Lemma B.1, $u(\mathbf{x}) > 0$ for p^{l}-almost every $\mathbf{x}$. Thus, we can restrict the expectation in Eq. (86) to the set $\{\mathbf{x} : u(\mathbf{x}) > 0\}$. On this set,

$$\left|\frac{\widetilde{r}(\mathbf{x})}{\widehat{u}(\mathbf{x})} - \frac{\eta(\mathbf{x})}{u(\mathbf{x})}\right| \le \frac{|\widetilde{r}(\mathbf{x}) - \eta(\mathbf{x})|}{\widehat{u}(\mathbf{x})} + \eta(\mathbf{x})\frac{|\widehat{u}(\mathbf{x}) - u(\mathbf{x})|}{u(\mathbf{x})\widehat{u}(\mathbf{x})}. \tag{87}$$

Since $\widehat{u}(\mathbf{x}) \ge \delta$ and $|\widetilde{r}(\mathbf{x}) - \eta(\mathbf{x})| \le \epsilon_r$ for p-almost every $\mathbf{x}$ satisfying $u(\mathbf{x}) > 0$, the expectation of the first term is bounded as

$$\mathbb{E}_{\mathbf{x}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\frac{|\widetilde{r}(\mathbf{x}) - \eta(\mathbf{x})|}{\widehat{u}(\mathbf{x})}\right] \le \frac{\epsilon_r}{\delta}. \tag{88}$$

For the second term, applying Lemma B.1 with

$$\varphi(\mathbf{x}) := \eta(\mathbf{x})\frac{|\widehat{u}(\mathbf{x}) - u(\mathbf{x})|}{\widehat{u}(\mathbf{x})}, \tag{89}$$

we obtain

$$\begin{aligned}
&\mathbb{E}_{\mathbf{x}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\eta(\mathbf{x})\frac{|\widehat{u}(\mathbf{x}) - u(\mathbf{x})|}{u(\mathbf{x})\widehat{u}(\mathbf{x})}\right] \\
&= \frac{1}{\alpha}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\eta(\mathbf{x})\frac{|\widehat{u}(\mathbf{x}) - u(\mathbf{x})|}{\widehat{u}(\mathbf{x})} I(\eta(\mathbf{x}) > 0)\right] \\
&\le \frac{\epsilon_u}{\alpha\delta}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\eta(\mathbf{x}) I(\eta(\mathbf{x}) > 0)\right] \\
&= \frac{\epsilon_u}{\alpha\delta}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\eta(\mathbf{x})] \\
&= \frac{\epsilon_u}{c\delta}.
\end{aligned} \tag{90}$$

Here, the inequality follows from $I(\eta(\mathbf{x}) > 0) = I(u(\mathbf{x}) > 0)$ p-almost surely in Lemma B.1, together with the assumptions on $\widehat{u}$. The last equality follows from $\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}[\eta(\mathbf{x})] = \pi$ and $\alpha = c\pi$. Combining Eqs. (86), (88), and (90), we obtain

$$\begin{aligned}
\sup_{s\in\mathcal{S}} |\mathcal{R}_\sigma(s;\widetilde{r},\widehat{u}) - \mathcal{R}_\sigma(s;\eta,u)| &\le \frac{c}{1-\pi}\left(\frac{\epsilon_r}{\delta} + \frac{\epsilon_u}{c\delta}\right) \\
&= \frac{c\epsilon_r + \epsilon_u}{(1-\pi)\delta}.
\end{aligned} \tag{91}$$

Let

$$s^* \in \arg\min_{s \in \mathcal{S}} \mathcal{R}_\sigma(s; \eta, u). \tag{92}$$

Using Eq. (91) for both $\widehat{s}$ and s^* and Eq. (85), we obtain

$$\begin{aligned}
\mathcal{R}_\sigma(\widehat{s}; \eta, u) &\leq \mathcal{R}_\sigma(\widehat{s}; \widetilde{r}, \widehat{u}) + \frac{c\epsilon_r + \epsilon_u}{(1-\pi)\delta} \\
&\leq \mathcal{R}_\sigma(s^*; \widetilde{r}, \widehat{u}) + 2\Gamma_{N^{\mathrm{p}},N}(\tau) + \xi_{\mathrm{opt}} + \frac{c\epsilon_r + \epsilon_u}{(1-\pi)\delta} \\
&\leq \mathcal{R}_\sigma(s^*; \eta, u) + 2\Gamma_{N^{\mathrm{p}},N}(\tau) + \xi_{\mathrm{opt}} + \frac{2(c\epsilon_r + \epsilon_u)}{(1-\pi)\delta},
\end{aligned} \tag{93}$$

which proves Eq. (81). □

The bound in Eq. (81) decomposes the excess smoothed AUC risk into the finite-sample statistical error, the estimation error of α, the confidence error, the labeling-probability estimation error, and the optimization error. For fixed finite function classes, $B_{N^{\mathrm{p}},N}(\tau)$ converges to zero as $N^{\mathrm{p}}, N \to \infty$. Thus, the excess risk converges to zero as the sample sizes increase, provided that the estimation errors of α, the confidence, and the labeling probability, as well as the optimization error, also vanish. Here, we consider finite function classes for a self-contained analysis. Extending the result to infinite function classes using Rademacher complexity is left for future work.

C Derivations for the Proposed AUC Risk

C.1 Constancy of the Positive-positive Term

Lemma C.1. $\mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \left[f(\mathbf{x}^{\mathrm{p}}, \bar{\mathbf{x}}^{\mathrm{p}}) \right] = \frac{1}{2}$.

Proof. Since $f(\mathbf{x}^{\mathrm{p}}, \bar{\mathbf{x}}^{\mathrm{p}}) = \sigma(-s(\mathbf{x}^{\mathrm{p}}) + s(\bar{\mathbf{x}}^{\mathrm{p}}))$ and $\sigma(z) + \sigma(-z) = 1$ for all $z \in \mathbb{R}$,

$$\begin{aligned}
\mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \left[f(\mathbf{x}^{\mathrm{p}}, \bar{\mathbf{x}}^{\mathrm{p}}) \right] &= \mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \left[1 - f(\bar{\mathbf{x}}^{\mathrm{p}}, \mathbf{x}^{\mathrm{p}}) \right] \\
&= 1 - \mathbb{E}_{\mathbf{x}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}} \sim p^{\mathrm{p}}(\mathbf{x})} \left[f(\bar{\mathbf{x}}^{\mathrm{p}}, \mathbf{x}^{\mathrm{p}}) \right].
\end{aligned} \tag{94}$$

It is clear that the lemma follows from this equation. □

C.2 Derivation of the Relation between the Labeled and Positive Densities

Lemma C.2. *Labeled density $p^{\mathrm{l}}(\mathbf{x}) = p(\mathbf{x}|o=1)$ can be represented as $p^{\mathrm{l}}(\mathbf{x}) = \frac{e(\mathbf{x})}{c} p^{\mathrm{p}}(\mathbf{x})$, where $e(\mathbf{x}) = p(o=1|\mathbf{x}, y=1)$, $c = p(o=1|y=1)$, and $p^{\mathrm{p}}(\mathbf{x}) = p(\mathbf{x}|y=1)$.*

Proof.

$$\begin{aligned}
p^{\mathrm{l}}(\mathbf{x}) &= p(\mathbf{x}|o=1) \\
&= p(\mathbf{x}|o=1, y=1) \\
&= \frac{p(\mathbf{x}, o=1, y=1)}{p(o=1, y=1)} \\
&= \frac{p(o=1|\mathbf{x}, y=1) p(\mathbf{x}|y=1) p(y=1)}{p(o=1|y=1) p(y=1)} \\
&= \frac{e(\mathbf{x})}{c} p^{\mathrm{p}}(\mathbf{x}),
\end{aligned} \tag{95}$$

where we used the PU property (labeled data are always positive) in the second equality and the Bayes' rule in the third equality. □

C.3 Relation between the Propensity Score and the Labeling Probability

Lemma C.3. *Labeling probability* $p(o = 1|\mathbf{x})$ *and propensity score* $e(\mathbf{x}) = p(o = 1|\mathbf{x}, y = 1)$ *satisfy*

$$p(o = 1|\mathbf{x}) = e(\mathbf{x})p(y = 1|\mathbf{x}). \tag{96}$$

Consequently, for any $\mathbf{x}$ *such that* $p(y = 1|\mathbf{x}) > 0$*,*

$$e(\mathbf{x}) = \frac{p(o = 1|\mathbf{x})}{p(y = 1|\mathbf{x})}. \tag{97}$$

Proof. By the law of total probability,

$$\begin{aligned} p(o = 1|\mathbf{x}) &= p(o = 1|\mathbf{x}, y = 1)p(y = 1|\mathbf{x}) + p(o = 1|\mathbf{x}, y = 0)p(y = 0|\mathbf{x}) \\ &= e(\mathbf{x})p(y = 1|\mathbf{x}), \end{aligned} \tag{98}$$

where the second equality follows from the definition of the propensity score and the PU assumption $p(o = 1|\mathbf{x}, y = 0) = 0$. Dividing both sides by $p(y = 1|\mathbf{x})$ gives Eq. (97). Moreover, since $p^{\mathrm{p}}(\mathbf{x}) = p(y = 1|\mathbf{x})p(\mathbf{x})/\pi$, we have $p(y = 1|\mathbf{x}) > 0$ for p^{p}-almost every $\mathbf{x}$. Therefore, Eq. (14) holds for p^{p}-almost every $\mathbf{x}$. □

D Extensions of the Proposed Method

D.1 Extension to Non-symmetric Loss Functions

In this section, we derive the AUC risk estimator when using non-symmetric function σ, which also enables us to maximize the AUC without requiring π and c.

Specifically, since the second term in Eq. (11) does not become a constant with non-symmetric functions, we need to calculate this term with biased positive data. Using Eqs. (13) and (14), as in the derivation of the proposed risk in the main paper, the second term in Eq. (11) can be rewritten as

$$\begin{aligned} &\frac{\pi}{1-\pi}\mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{p}}(\mathbf{x})}\mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}}\sim p^{\mathrm{p}}(\mathbf{x})}\left[f(\mathbf{x}^{\mathrm{p}}, \bar{\mathbf{x}}^{\mathrm{p}})\right] \\ &\quad = \frac{c\alpha}{1-\pi}\mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\frac{p(y = 1|\mathbf{x}^{\mathrm{p}})}{p(o = 1|\mathbf{x}^{\mathrm{p}})}\frac{p(y = 1|\bar{\mathbf{x}}^{\mathrm{p}})}{p(o = 1|\bar{\mathbf{x}}^{\mathrm{p}})}f(\mathbf{x}^{\mathrm{p}}, \bar{\mathbf{x}}^{\mathrm{p}})\right], \end{aligned} \tag{99}$$

where we used $\alpha = c\pi$.

By replacing the constant term $\frac{\pi}{2(1-\pi)}$ in Eq. (16) with the above-derived term, we can obtain the AUC risk with non-symmetric functions as follows:

$$\begin{aligned} \mathcal{R}(s) = \frac{c}{1-\pi}\Bigg[&\alpha\mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\frac{p(y = 1|\mathbf{x}^{\mathrm{p}})}{p(o = 1|\mathbf{x}^{\mathrm{p}})}f(\mathbf{x}^{\mathrm{p}}, \bar{\mathbf{x}}^{\mathrm{p}})\right] \\ &+(1-\alpha)\mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\mathbb{E}_{\mathbf{x}\sim p^{\mathrm{u}}(\mathbf{x})}\left[\frac{p(y = 1|\mathbf{x}^{\mathrm{p}})}{p(o = 1|\mathbf{x}^{\mathrm{p}})}f(\mathbf{x}^{\mathrm{p}}, \mathbf{x})\right] \\ &-\alpha\mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\mathbb{E}_{\bar{\mathbf{x}}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\left[\frac{p(y = 1|\mathbf{x}^{\mathrm{p}})}{p(o = 1|\mathbf{x}^{\mathrm{p}})}\frac{p(y = 1|\bar{\mathbf{x}}^{\mathrm{p}})}{p(o = 1|\bar{\mathbf{x}}^{\mathrm{p}})}f(\mathbf{x}^{\mathrm{p}}, \bar{\mathbf{x}}^{\mathrm{p}})\right]\Bigg]. \end{aligned} \tag{100}$$

The corresponding AUC risk estimator can be obtained by approximating the expectation with the sample average. Since the coefficient $c/(1-\pi)$ does not affect the optimization of s, we can also perform training without knowing π and c.

D.2 Extension to the Two-sample Scenario

In the main paper, we assume the one-sample scenario for collecting PU data. In this section, we consider the proposed method for the two-sample scenario. In this scenario, PU data are sampled

independently: labeled positive data are drawn from $p^{\mathrm{l}}(\mathbf{x})$ and unlabeled data are drawn from true marginal density $p(\mathbf{x})$. As a result, we are given the following data for training:

$$\begin{aligned} X^{\mathrm{p}} &:= \{\mathbf{x}_n^{\mathrm{p}}\}_{n=1}^{N^{\mathrm{p}}} \sim p^{\mathrm{l}}(\mathbf{x}) := p(\mathbf{x}|o=1), \\ X &:= \{\mathbf{x}_m\}_{m=1}^{N} \sim p(\mathbf{x}), \\ R^{\mathrm{p}} &:= \{r_n^{\mathrm{p}}|\ r_n^{\mathrm{p}} = p(y=1|\mathbf{x}_n^{\mathrm{p}})\}_{n=1}^{N^{\mathrm{p}}}. \end{aligned} \tag{101}$$

In this scenario, the relationship between $p^{\mathrm{l}}(\mathbf{x})$ and $p^{\mathrm{p}}(\mathbf{x})$ in Eq. (13) is also satisfied. Additionally, propensity score $e(\mathbf{x})$ can be rewritten as

$$\begin{aligned} e(\mathbf{x}) &= p(o=1|\mathbf{x}, y=1) \\ &= \frac{p(o=1, \mathbf{x}, y=1)}{p(\mathbf{x}, y=1)} \\ &= \frac{p(y=1|\mathbf{x}, o=1)p(\mathbf{x}|o=1)p(o=1)}{p(y=1|\mathbf{x})p(\mathbf{x})} \\ &= \frac{p(\mathbf{x}|o=1)p(o=1)}{p(y=1|\mathbf{x})p(\mathbf{x})} \\ &= \frac{\alpha w(\mathbf{x})}{p(y=1|\mathbf{x})}, \end{aligned} \tag{102}$$

where we used the Bayes' rule in the second equality, the PU property (labeled data are always positive), $p(y=1|\mathbf{x}, o=1)=1$, in the fourth equality, and set $w(\mathbf{x}) := \frac{p(\mathbf{x}|o=1)}{p(\mathbf{x})}$ and $\alpha := p(o=1)$ in the fifth equality. Here, unlike Eq. (14) of the one-sample scenario, we do not use $p(o=1|\mathbf{x})$ to represent $e(\mathbf{x})$. This is because the numbers of PU data are arbitrarily determined by the users (data collectors) in the two-sample scenario and thus true label ratio α cannot be estimated from the given data (Elkan & Noto, 2008). As a result, $p(o=1|\mathbf{x}) = \alpha p(\mathbf{x}|o=1)/p(\mathbf{x})$ cannot also be estimated from the given data (Elkan & Noto, 2008). In contrast, density-ratio $w(\mathbf{x}) = p(\mathbf{x}|o=1)/p(\mathbf{x})$ in Eq. (102) can be estimated from given PU data without knowing α by using off-the-shelf density-ratio estimation methods (Kanamori et al., 2009; Kato & Teshima, 2021; Kato et al., 2019; Sugiyama et al., 2012).

By substituting Eqs. (13) and (102) into Eq. (12), AUC risk $R_\sigma(s)$ can be represented as

$$\mathcal{R}_\sigma(s) = \frac{c}{(1-\pi)\alpha}\mathbb{E}_{\mathbf{x}^{\mathrm{p}}\sim p^{\mathrm{l}}(\mathbf{x})}\mathbb{E}_{\mathbf{x}\sim p(\mathbf{x})}\left[\frac{p(y=1|\mathbf{x}^{\mathrm{p}})}{w(\mathbf{x}^{\mathrm{p}})}f(\mathbf{x}^{\mathrm{p}}, \mathbf{x})\right] - \frac{\pi}{2(1-\pi)}. \tag{103}$$

Here, coefficient $c/((1-\pi)\alpha)$ and constant $\pi/(2(1-\pi))$ do not affect the optimization for score function s and thus we can safely ignore them. The estimator of the AUC risk in Eq. (103) with training data $X^{\mathrm{p}} \cup X \cup R^{\mathrm{p}}$ can be represented as

$$\hat{\mathcal{R}}_\sigma(s) = \frac{c}{(1-\pi)\alpha}\left[\frac{1}{N^{\mathrm{p}}N}\sum_{n,m=1}^{N^{\mathrm{p}},N}\frac{r_n^{\mathrm{p}}}{\hat{w}_n^{\mathrm{p}}}f(\mathbf{x}_n^{\mathrm{p}}, \mathbf{x}_m)\right] - \frac{\pi}{2(1-\pi)}, \tag{104}$$

where $\hat{w}_n^{\mathrm{p}}$ denotes the estimate of density-ratio $w(\mathbf{x}_n^{\mathrm{p}})$ for instance $\mathbf{x}_n^{\mathrm{p}}$.

E Additional Experimental Details

E.1 Datasets and Data Construction

Mnist consists of hand-written images of 10 digits. Fmnist consists of images of 10 fashion categories. In Mnist and Fmnist, each image is represented by gray scale with 28×28 pixels. Svhn consists of images of 10 printed digits clipped from photographs of house number plates. Cifar10 consists of images of 10 animal and vehicle categories. In Svhn and Cifar10, each image is represented by 32×32 RGB pixels. Diabetes is a tabular dataset whose task is to predict whether the respondent has diabetes or not. Each respondent is represented as 142-dimensional features. Blood is a tabular dataset whose task is hypertension diagnosis for high-risk age people. Each survey subject is represented as 100-dimensional features.

We created the binary classification problems from each dataset following the data creation procedure of previous studies (Kumagai et al., 2025b; 2024; 2025a; Xie et al., 2024). Specifically, for Mnist and Svhn, we used odd digits as positive and even digits as negative. For Fmnist, we used non-upper garment classes (1, 5, 7, 8, and 9) as positive and the others as negative. For Cifar10, we used the vehicle classes (0, 1, 8, and 9) as positive and the animal classes as negative. For Diabetes and Blood, we used the original binary class labels.

To create the sampling bias, for Mnist and Svhn, 90% of data were selected from digits 1, 3, and 5, and 10% of data were selected from digits 7 and 9 for labeled positive data. For Fmnist, 90% of data were selected from classes 1, 5, and 7, and 10% of data were selected from classes 8 and 9 for labeled positive data. For Cifar10, 90% of data were selected from classes 0, and 1, and 10% of data were selected from classes 8 and 9 for labeled positive data. The remaining unselected data were used for unlabeled data for training and validation.

To create the selection bias in Diabetes and Blood datasets, we followed the procedure used in previous studies (Sakai & Shimizu, 2019; Kumagai et al., 2025a). Specifically, we first calculated the Euclidean distance between positive instance $\mathbf{x}_n$ and the mean vector of all positive data $\bar{\mathbf{x}}$, $\mathbf{c}_n := \|\mathbf{x}_n - \bar{\mathbf{x}}\|$. Then, we found the median $\mathbf{c}_{\mathrm{med}}$ from all $\{\mathbf{c}_n\}$. We split all $\{\mathbf{c}_n\}$ into the first set whose elements were smaller than $\mathbf{c}_{\mathrm{med}}$ and the second set whose elements were greater than or equal to $\mathbf{c}_{\mathrm{med}}$. Within labeled positive data, 90% data were selected from the first set and 10% data from the second set. In addition, we provide the results when using other selection biases for the image and tabular datasets in Section F.4.

E.2 Datasets with Human-derived Confidence

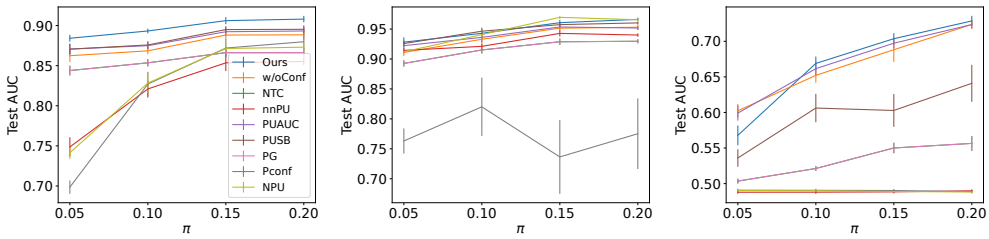
Cifar10-H and Fmnist-H were real-world image datasets where each instance was labeled by approximately 50 and 67 real-world human annotators, respectively (Peterson et al., 2019; Ishida et al., 2023). In these datasets, confidence information for each instance can be obtained by averaging the corresponding annotations (Ishida et al., 2023). For Cifar10-H, we treated the land-related classes (1, 3, 4, 5, 7, and 9) as positive and the remaining classes (e.g., water- or sky-related classes) as negative following the previous study (Ishida et al., 2023). To create the sampling bias, 90% of the labeled positive data were sampled from classes 1, 3, and 4. For Fmnist-H, we treated the casual classes (0, 1, 2, 6, and 7) as positive and the non-casual classes (3, 4, 5, 8, and 9) as negative. To create the sampling bias, 90% of the labeled positive data were sampled from classes 0, 1, and 2. Since the amount of annotated data was limited, we set the number of initially sampled unlabeled training to $3,000$ for both datasets. The other conditions were the same as those used for Cifar10 and Fmnist.

E.3 Model and Training Details

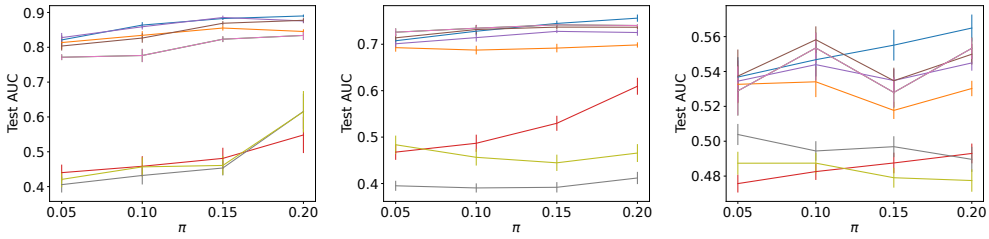
For Mnist, Fmnist, Fmnist-H, Diabetes, and Blood, a three-layered feed-forward neural network with ReLU activation was used for the classifier (score function). The number of hidden nodes was 124. For Svhn, Cifar10, and Cifar10-H, a convolutional neural network, which consisted of two convolutional blocks followed by a three-layered feed-forward neural network, was used for the classifier (score function). The first (second) convolutional block comprised a 6 (16) filter 5×5 convolution, the ReLU activation, and a 2×2 max-pooling layer. For all comparison methods including the classifier to estimate positive-confidence, the same neural network architecture was used.

For the proposed method, w/oConf, nnPU, PUAUC, Pconf, and NPU, the sigmoid function was used as a surrogate, following the previous studies (Kiryo et al., 2017; Kumagai et al., 2024; 2025a). For PUSB, we used the logarithmic loss function to deal with the selection bias following the original paper (Kato et al., 2019). For NTC and the classifier to estimate $p(y = 1|\mathbf{x})$, we used the logistic regression with the neural network as probabilistic classifiers. For fair comparison, we adapted nnPU, PUAUC, PUSB, and NPU to the one-sample setting by exactly rewriting expectations over the marginal density using the labeled positive and unlabeled densities (Bekker & Davis, 2020).

For all methods, the empirical risk (loss) with validation PU data was used for early-stopping to mitigate overfitting. For NPU, the weighting parameter was also chosen on the basis of the validation empirical risk from $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$. Confidence score $\tilde{r}(\mathbf{x})$ used in NPU



(a) Mnist (b) Fmnist (c) Svhn



(d) Cifar10 (e) Diabetes (f) Blood

Figure 3: Average test AUCs with their standard errors when changing positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$.

was set to one since the observed positive labels are noise-free in our setting. For the proposed method and w/oConf, we rounded up $\hat{u}_n^{\mathrm{p}} = \hat{u}(\mathbf{x}_n^{\mathrm{p}})$ in Eq. (17) less than 0.01 to 0.01 to stabilize the training process following the previous study (Ishida et al., 2018). Similarly, for Pconf and NPU, we rounded up $r_n^{\mathrm{p}} = p(y = 1|\mathbf{x}_n^{\mathrm{p}})$ less than 0.01 to 0.01 for their objectives. For all methods, we used the Adam optimizer (Kingma & Ba, 2014). We set the learning rate to 10^{-4}. Total mini-batch size $M = U + P$ was set to $1,024$ and PU mini-batch sizes P and U were set so as to maintain the label ratio α in the given all data. For the classifier for estimating $p(y = 1|\mathbf{x})$, we set the training epoch to 30. For other methods, the maximum number of epochs was 200 and the early-stopping was used. All methods were implemented using PyTorch (Paszke et al., 2017), and all experiments were conducted on a Linux server with an Intel Xeon CPU and A100 GPU.

F Additional Experimental Results

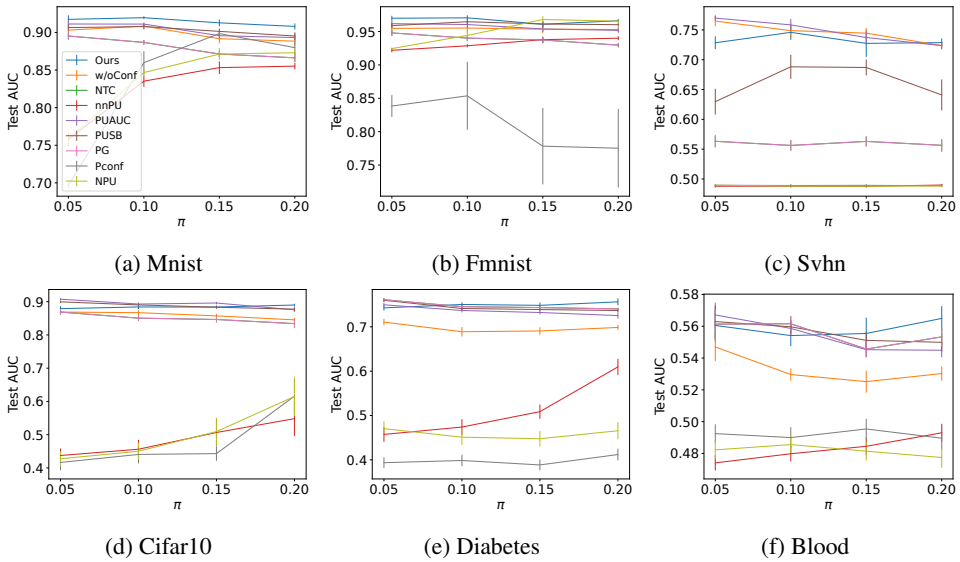
F.1 Results with Different Positive Class-priors π

Figure 3 shows the average test AUCs with their standard errors when changing the value of positive class-prior π. The proposed method tended to perform well with each π value. Most methods tended to improve the performance as the value of π increased. This is because the number of labeled positive data increased when π increased due to the relationship $\alpha = p(o = 1) = c\pi$ with fixed $c = p(o = 1|y = 1)$.

Figure 4 shows the average test AUCs with their standard errors when changing the value of positive class-prior π while maintaining the labeled positive data size. To maintain the size, we adjusted the value of $c = p(o = 1|y = 1)$ such that $p(o = 1) = c\pi = 0.02$ for each π. As a result, the labeled positive data and unlabeled data sizes were 100 and $4,900$, respectively. The proposed method worked well in most cases and simply increasing π did not lead to performance improving.

F.2 Results with Different Positive Labeling Rates $c = p(o = 1|y = 1)$

Tables 3 and 4 show the average test AUCs over different positive class-prior π when changing the value of $c = p(o = 1|y = 1)$. The proposed method performed well for each c value. As c increases, the number of labeled positive data increases, and thus the performance of the proposed method tends to improve.



(a) Mnist (b) Fmnist (c) Svhn

(d) Cifar10 (e) Diabetes (f) Blood

Figure 4: Average test AUCs with their standard errors when changing positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$ with the fixed number of labeled positive data.

Table 3: Average test AUCs over different positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$ with $c = 0.05$.

Data	Ours	w/oConf	NTC	nnPU	PUAUC	PUSB	PG	Pconf	NPU
Mnist	**0.8643**	0.8465	0.8136	0.8009	0.8538	0.8525	0.8136	0.7697	0.8095
Fmnist	**0.9220**	0.9141	0.8827	0.9205	0.9164	**0.9305**	0.8827	0.6691	**0.9318**
Svhn	**0.5896**	**0.5826**	0.5090	0.4885	**0.5911**	0.5465	0.5090	0.4906	0.4898
Cifar10	**0.8235**	0.7974	0.7312	0.4751	**0.8151**	0.7979	0.7312	0.4976	0.4838
Diabetes	**0.7109**	0.6821	0.6949	0.5660	0.6905	**0.7065**	0.6949	0.4013	0.4744
Blood	**0.5311**	0.5170	0.5144	0.4905	0.5196	**0.5269**	0.5144	0.5041	0.4780
# Best	6	1	0	0	2	3	0	0	1

Table 4: Average test AUCs over different positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$ with $c = 0.15$.

Data	Ours	w/oConf	NTC	nnPU	PUAUC	PUSB	PG	Pconf	NPU
Mnist	**0.9131**	0.8931	0.8712	0.8216	0.8978	0.8972	0.8712	0.8456	0.8323
Fmnist	**0.9589**	0.9475	0.9333	0.9317	0.9512	0.9544	0.9333	0.8156	**0.9529**
Svhn	**0.7095**	**0.7185**	0.5524	0.4882	**0.7167**	0.6493	0.5524	0.4891	0.4884
Cifar10	0.8804	0.8548	0.8350	0.4949	**0.8882**	0.8751	0.8350	0.4867	0.4902
Diabetes	**0.7478**	0.7000	**0.7483**	0.5032	0.7321	**0.7439**	**0.7483**	0.3956	0.4628
Blood	**0.5609**	0.5314	0.5489	0.4829	0.5431	0.5495	0.5489	0.4947	0.4826
# Best	5	1	1	0	2	1	1	0	1

F.3 VISUALIZATION

The proposed method assigns weight $p(y = 1|\mathbf{x})/p(o = 1|\mathbf{x})$ to the labeled positive instance $\mathbf{x}$ in the AUC risk of Eq. (16) for dealing with the bias of given positive data. In this section, we investigated the distribution of the estimated weights for labeled positive data with the Mnist dataset. Figure 5 shows the result. As can be seen, the values of the weights ranged from around 0.05 to about 84.0, with the majority distributed between 0.0 and 10.0. By leveraging these estimated weights, the proposed method performed superiorly.

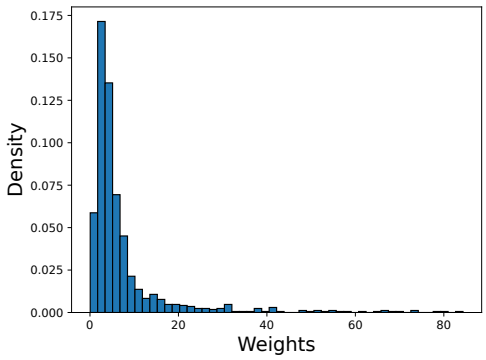


Figure 5: Weight distribution of the proposed method with the Mnist dataset when $\pi = 0.2$.

Table 5: The results using confidence-dependent selection bias: average test AUCs over different positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$. Values in bold are not statistically different at the 5% level from the best performing method in each row according to a paired t-test.

Data	Ours	w/oConf	NTC	nnPU	PUAUC	PUSB	PG	Pconf	NPU
Mnist	**0.9142**	0.9063	0.8941	0.8876	0.9132	**0.9162**	0.8941	0.7540	0.8901
Fmnist	**0.9748**	0.9728	0.9488	0.9237	**0.9738**	**0.9709**	0.9488	0.4847	0.9494
Svhn	**0.6546**	**0.6594**	0.5290	0.4890	**0.6650**	0.5831	0.5290	0.4899	0.4894
Cifar10	**0.8614**	0.8499	0.8103	0.5196	**0.8595**	0.8444	0.8103	0.5608	0.5788
Diabetes	**0.7684**	0.7395	**0.7682**	0.5923	0.7643	**0.7699**	**0.7682**	0.3808	0.5249
Blood	**0.6129**	0.5846	0.5927	0.5011	0.6076	**0.6085**	0.5927	0.4769	0.4762

Table 6: Training time [sec] of each method on the Mnist dataset.

Ours	w/oConf	NTC	nnPU	PUAUC	PUSB	PG	Pconf	NPU
84.5424	84.6170	84.8195	84.5650	83.3238	85.2143	0.3280	80.4764	84.9054

F.4 Results with Confidence-dependent Selection Bias

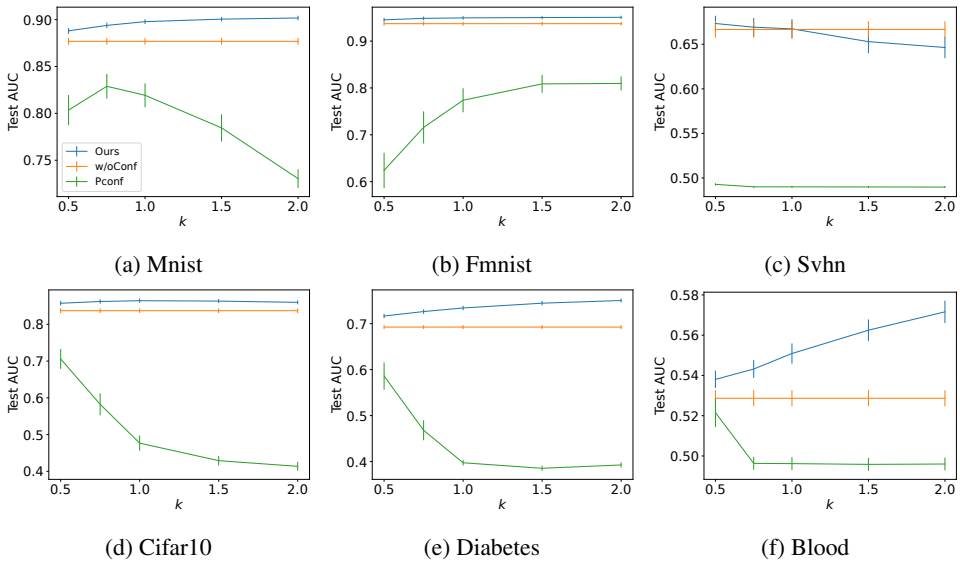
In the main paper, we considered class-based and distance-based selection biases for the image and tabular datasets, respectively. In this section, we evaluated the proposed method using a confidence-dependent selection bias, which often arises in real-world applications. Specifically, we fixed the fraction of labeled positive instances at $c = 0.1$, so that $N^{\mathrm{P}} = Vc\pi$, and sampled these instances from the positive instances in an initial data pool of size V, using sampling weights proportional to $\sigma(4r(\mathbf{x}) - 2)$, where $r(\mathbf{x}) = p(y = 1|\mathbf{x})$. We used the probabilistic classifier for estimating $r(\mathbf{x})$. In this setting, positive instances with higher confidence were more likely to be selected as labeled data. Table 5 shows the results. The proposed method performed well under this confidence-dependent selection bias.

F.5 Computation Costs

We investigated the training time of the proposed method on the Mnist dataset. We used a Linux server with a 2.20GHz CPU. Table 6 shows the results. Although the proposed method (Ours), w/oConf, and PG require the training of $p(o = 1|\mathbf{x})$, we omitted its training time here because we wanted to purely investigate the computation cost of our proposed loss function in Eq. (17). There were no significant differences in training time among all methods except for PG. The training time of PG is short since the classifier of PG can be obtained analytically from trained $p(o = 1|\mathbf{x})$. However, it had lower test AUCs than the proposed method as described in Table 1. The full training times, including the training time of $p(o = 1|\mathbf{x})$, for the proposed method (Ours), w/oConf, and PG were 169.3620, 169.4360, and 85.1475, respectively. Although the proposed method had a longer training time than other methods due to the additional training of $p(o = 1|\mathbf{x})$, it had superior test AUCs.

Table 7: Comparison with LBE: average test AUCs over different positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$. We set $c = p(o = 1|y = 1) = 0.1$. Values in bold are not statistically different at the 5% level from the best performing method in each column according to a paired t-test.

Method	Mnist	Fmnist	Svhn	Cifar10	Diabetes	Blood
Ours	**0.8979**	**0.9495**	**0.6673**	**0.8643**	**0.7342**	**0.5509**
LBE	0.8631	0.9245	0.5668	0.8271	**0.7292**	0.5399



(a) Mnist (b) Fmnist (c) Svhn

(d) Cifar10 (e) Diabetes (f) Blood

Figure 6: The average test AUCs and their standard errors over different positive class-priors for different parameters k of the transformation $h(r) = r^k$.

F.6 Comparison with an Additional SAR-specific Method

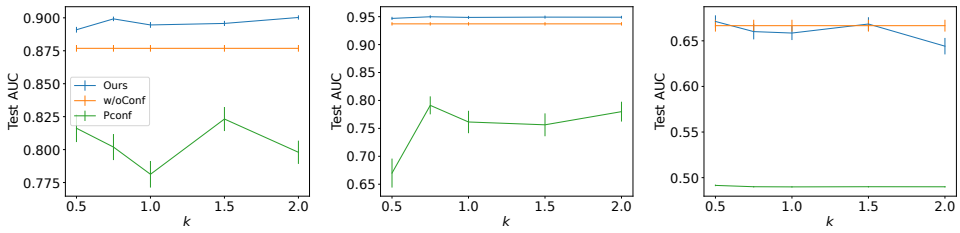
To further compare the proposed method with methods specifically designed for the SAR setting, we additionally evaluated LBE (LBE-MLP) (Gong et al., 2021), a representative PU learning method for both the SAR setting and the one-sample scenario. This method estimates both $p(y = 1|\mathbf{x})$ and $p(o = 1|\mathbf{x}, y = 1)$ from biased PU data using the EM algorithm. Table 7 shows the results. The proposed method often outperformed LBE. This would be because LBE is not designed for AUC maximization and has difficulty identifying $p(o = 1|\mathbf{x}, y = 1)$ from biased PU data. In contrast, our method is designed to maximize AUC and to avoid the identifiability issue by using positive-confidence.

F.7 Results under Strictly Increasing Transformations of Positive-confidence

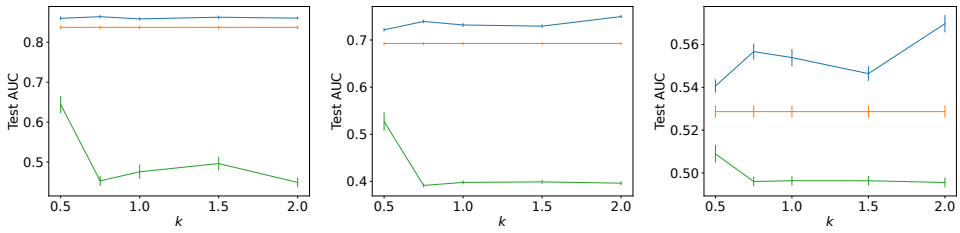
Figures 6 and 7 show the average test AUCs with their standard errors when changing k of the transformations $h(r) = r^k$ and $h(r) = r^k/(r^k + (1 - r)^k)$, respectively. For both transformation forms, the performance of the proposed method remained relatively stable across different k values and showed no large degradation as k moved away from 1. In contrast, the performances of Pconf that also uses positive-confidence substantially degraded for some values of k, reflecting its sensitivity to transformations of the confidence values.

F.8 Results with Gaussian Noise on Positive-confidence

As shown in Section 4.3, the proposed method can recover the Bayes-optimal AUC ranking even when the positive-confidence is distorted by a strictly increasing transformation. Here, we evaluated its robustness when random noise was added to the positive-confidence. For noisy



(a) Mnist (b) Fmnist (c) Svhn



(d) Cifar10 (e) Diabetes (f) Blood

Figure 7: The average test AUCs and their standard errors over different positive class-priors for different parameters k of the transformation $h(r) = r^k/(r^k + (1-r)^k)$.

Table 8: The results with noisy positive-confidence: average test AUCs over different positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$. Std represents the standard deviation of Gaussian noise. $\mathrm{Std} = 0.0$ means that there is no noise.

Std	0.0	0.01	0.05	0.1	0.15	w/oConf
Mnist	0.8979	0.8966	0.8932	0.8905	0.8883	0.8769
Fmnist	0.9495	0.9495	0.9501	0.9496	0.9489	0.9371
Svhn	0.6673	0.6662	0.6601	0.6367	0.6161	0.6666
Cifar10	0.8643	0.8670	0.8623	0.8579	0.8552	0.8371
Diabetes	0.7342	0.7333	0.7277	0.7189	0.7013	0.6926
Blood	0.5509	0.5457	0.5435	0.5376	0.5321	0.5287

positive-confidence, we added the zero-mean Gaussian noise with standard deviations chosen from $\{0.01, 0.05, 0.1, 0.15\}$ following the previous study (Ishida et al., 2018). As the standard deviation increases, the noise on positive-confidence becomes bigger. When the modified positive-confidence was less than 0 or greater than 1, we clipped the value to 0 and 1, respectively. Table 8 shows the results. As expected, the performance of the proposed method tended to decrease as the noise increased, because our objective function in Eq. (17) explicitly depends on confidence values. However, the degradation was relatively small on several datasets. Even with a standard deviation of 0.15, the proposed method often outperformed w/oConf, which does not use positive-confidence. These results suggest that using positive-confidence can remain beneficial even when the confidence contains random noise.

F.9 Robustness analysis for the Clipping

As mentioned in Section 5.2, we used the clipping for estimated labeling probability $\hat{u}(\mathbf{x})$ to stabilize the training process. Specifically, when the estimated labeling probability was smaller than a clipping value, we replaced it with the clipping value. Here, we empirically investigated the robustness of the proposed method to the clipping values. Table 9 shows the average test AUCs by changing clipping value τ within $\{0.001, 0.005, 0.01, 0.05, 0.1\}$. These results show that the proposed method is relatively robust against the choice of the clipping values.

Table 9: Average test AUCs when varying the clipping value τ for the estimated labeling probability $p(o = 1|\mathbf{x})$.

Data	$\tau = 0.001$	0.005	0.01	0.05	0.1
Mnist	0.8945	0.8962	0.8979	0.8997	0.8995
Fmnist	0.9467	0.9480	0.9495	0.9503	0.9494
Svhn	0.6656	0.6672	0.6673	0.6525	0.6533
Cifar10	0.8605	0.8624	0.8643	0.8642	0.8631
Diabetes	0.7325	0.7322	0.7342	0.7392	0.7400
Blood	0.5480	0.5493	0.5509	0.5564	0.5564

Table 10: The results with noisy estimated labeling probabilities: average test AUCs over different positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$. Std represents the standard deviation of Gaussian noise. $\mathrm{Std} = 0.0$ means that there is no noise.

Std	0.0	0.01	0.05	0.1	0.15
Mnist	0.8979	0.8961	0.8933	0.8896	0.8867
Fmnist	0.9495	0.9501	0.9468	0.9403	0.9364
Svhn	0.6673	0.6595	0.6354	0.6214	0.6166
Cifar10	0.8643	0.8689	0.8621	0.8567	0.8547
Diabetes	0.7342	0.7351	0.7325	0.7329	0.7328
Blood	0.5509	0.5453	0.5421	0.5412	0.5401

F.10 Results with Gaussian Noise on Estimated Labeling Probability

The proposed method uses the labeling probability estimated from the given PU data, $\hat{u}(\mathbf{x})$, to construct the rewritten AUC estimator. Here, we evaluated the proposed method when random noise was added to estimated labeling probability $\hat{u}(\mathbf{x})$. Specifically, we added the zero-mean Gaussian noise with standard deviations chosen from $\{0.01, 0.05, 0.1, 0.15\}$ to $\hat{u}(\mathbf{x})$. When the modified labeling probability was less than 0.01 or greater than 1, we clipped the value to 0.01 and 1, respectively. Table 10 shows the results. As expected, the performance of the proposed method tended to decrease as the noise increased. However, the degradation was relatively small on several datasets. Even with a standard deviation of 0.15, the proposed method often worked well. These results empirically demonstrate that the proposed method is reasonably robust to estimation errors in the labeling probability.

F.11 Full Results

Table 11 shows the average test AUCs with their standard deviations over different positive class-prior π in each dataset. The proposed method performed the best or comparably to it in all cases.

Table 11: Average test AUCs with their standard deviations over different positive class-prior π within $\{0.05, 0.1, 0.15, 0.2\}$. We set $c = p(o = 1|y = 1) = 0.1$. Values in bold are not statistically different at the 5% level from the best performing method in each row according to a paired t-test.

Data	Ours	w/oConf	NTC	nnPU	PUAUC
Mnist	**0.8979(0.0158)**	0.8769(0.0226)	0.8575(0.0192)	0.8196(0.0531)	0.8828(0.0203)
Fmnist	**0.9495(0.0236)**	0.9371(0.0267)	0.9165(0.0219)	0.9296(0.0180)	0.9409(0.0231)
Svhn	**0.6673(0.0688)**	**0.6666(0.0579)**	0.5328(0.0305)	0.4885(0.0069)	**0.6704(0.0539)**
Cifar10	**0.8643(0.0373)**	0.8371(0.0420)	0.8013(0.0491)	0.4819(0.1191)	**0.8620(0.0355)**
Diabetes	**0.7342(0.0306)**	0.6926(0.0268)	**0.7357(0.0183)**	0.5234(0.0775)	0.7171(0.0258)
Blood	**0.5509(0.0317)**	0.5287(0.0253)	0.5409(0.0336)	0.4847(0.0182)	0.5395(0.0273)

Data	PUSB	PG	Pconf	NPU
Mnist	0.8841(0.0197)	0.8575(0.0192)	0.8193(0.0807)	0.8285(0.0598)
Fmnist	**0.9474(0.0195)**	0.9165(0.0219)	0.7737(0.1616)	**0.9470(0.0276)**
Svhn	0.5965(0.0763)	0.5328(0.0305)	0.4900(0.0070)	0.4893(0.0068)
Cifar10	0.8442(0.0446)	0.8013(0.0491)	0.4767(0.1310)	0.4883(0.1395)
Diabetes	0.7230(0.0248)	**0.7357(0.0183)**	0.3974(0.0368)	0.4626(0.0605)
Blood	**0.5450(0.0323)**	0.5409(0.0336)	0.4962(0.0204)	0.4828(0.0200)