When All We Need is a Piece of the Pie: A Generic Framework for Optimizing Two-way Partial AUC

Zhiyong Yang [1 2] **Qianqian Xu** [3] **Shilong Bao** [1 2] **Yuan He** [4] **Xiaochun Cao** [1 2] **Qingming Huang** [3 5 6 7]

Abstract

The Area Under the ROC Curve (AUC) is a crucial metric for machine learning, which evaluates the average performance over all possible True Positive Rates (TPRs) and False Positive Rates (FPRs). Based on the knowledge that a skillful classifier should simultaneously embrace a high TPR and a low FPR, we turn to study a more general variant called Two-way Partial AUC (TPAUC), where only the region with $\mathsf{TPR} \geq \alpha, \mathsf{FPR} \leq \beta$ is included in the area. Moreover, a recent work shows that the TPAUC is essentially inconsistent with the existing Partial AUC metrics where only the FPR range is restricted, opening a new problem to seek solutions to leverage high TPAUC. Motivated by this, we present the first trial in this paper to optimize this new metric. The critical challenge along this course lies in the difficulty of performing gradient-based optimization with end-to-end stochastic training, even with a proper choice of surrogate loss. To address this issue, we propose a generic framework to construct surrogate optimization problems, which supports efficient end-to-end training with deep-learning. Moreover, our theoretical analyses show that: 1) the objective function of the surrogate problems will achieve an upper bound of the original problem under mild conditions, and 2) optimizing the surrogate problems leads to good generalization performance in terms of TPAUC with a high probability. Finally, empirical studies over several benchmark datasets speak to the efficacy of our framework.

[1]State Key Laboratory of Info. Security (SKLOIS), Inst. of Info. Engin., CAS, Beijing, China. [2]School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China. [3]Key Lab of Intell. Info. Process., Inst. of Comput. Tech., CAS, Beijing, China. [4]Alibaba Group, Beijing, China. [5]School of Computer Science and Tech., University of Chinese Academy of Sciences, Beijing, China. [6]Peng Cheng Laboratory, Shenzhen, China. [7]BDKM, University of Chinese Academy of Sciences, Beijing, China. Correspondence to: Qianqian Xu <xuqianqian@ict.ac.cn>, Qingming Huang <qmhuang@ucas.ac.cn>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

1. Introduction

ROC (Receiver Operating Characteristics) curve is a well-known tool to evaluate classification performance at varying threshold levels. More precisely, as shown in Fig.1-(a), it captures the relationship between True Positive Rate (TPR) and False Positive Rate (FPR) as a function of the classification thresholds. AUC (Area Under the ROC Curve), summarizes the average performance of a given classifier by calculating its area. More intuitively, as shown in (Hanley & McNeil, 1982), AUC is equivalent to the possibility that a positive instance has a higher predicted score to be positive than a negative instance. Any skillful classifier that can produce well-separated scores for positive and negative instances will enjoy a high AUC value, no matter how skewed the class distribution is. As a natural result, AUC is more appropriate than accuracy for long-tail classification problems such as disease prediction (Hao et al., 2020; Zhou et al., 2020) and rare event detection (Liu et al., 2018; Wu et al., 2020; Liu et al., 2020a; Wang et al., 2019), due to this appealing property (Fawcett, 2006; Hand & Till, 2001).

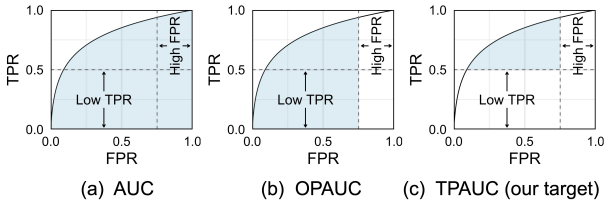


Figure 1. Comparisons of different AUC variants: **(a)** The entire area of ROC curve; **(b)** The One-way Partial AUC (OPAUC) which measures the area of a local region of ROC within an FPR range; **(c)** The Two-way Partial AUC (TPAUC).

Over the past two decades, the importance of AUC has raised a new wave to directly optimize AUC, which has achieved tremendous success. A partial list of the related studies includes (Alan & Raskutti, 2004; Joachims, 2005; 2006; Calders & Jaroszewicz, 2007; Narasimhan & Agarwal, 2013a; Gao et al., 2013; Narasimhan & Agarwal, 2017a).

However, the vast majority of such studies only consider the area over the entire ROC curve. As argued by (Narasimhan & Agarwal, 2013b), for some applications, only the performance within a given range of False Positive Rate (FPR) is of interest, as shown in Fig.1-(b). In this sense, the standard AUC tends to provide a biased estimation of the performance by including unrelated regions. This key investigation has motivated a series of successful studies to optimize the One-way Partial AUC (OPAUC) with an FPR range $[\alpha, \beta]$ (Narasimhan & Agarwal, 2013b;c; 2017b). Here we note that the choice to truncate FPR on the ROC curve is based on domain-specific prior knowledge for some specific fields such as biometric screening, and medical diagnosis (Narasimhan & Agarwal, 2013b).

Taking a step further, what should be a general rule to select the target region under the ROC curve for classification problems?

Since TPR and FPR evaluate complementary aspects of the model performance, we argue that a practical classifier in most applications must simultaneously have a high TPR and a low FPR. In other words, a high TPR is meaningless if the FPR is lower than a tolerance threshold, while a low FPR cannot compensate for a low TPR (*say, one can hardly consider a model with FPR higher than 0.8 even if its TPR is as high as 0.99, and vice versa for a low TPR model*). In this sense, we only need to pay attention to the upper-left head region under the ROC curve, as shown in Fig.1-(c).

A recent work (Yang et al., 2019) exactly realizes this idea, where a new metric called Two-Way Partial AUC (TPAUC) is proposed to measure the area of a partial region of the ROC curve with $\mathsf{TPR} \geq p, \mathsf{FPR} \leq q$. Furthermore, (Yang et al., 2019) shows that the TPAUC is essentially inconsistent with one-way partial AUC. In other words, a higher OPAUC does not necessarily imply a higher TPAUC, posing a demand to seek new solutions to leverage high TPAUC.

Inspired by this fact, we present the first trial to optimize the TPAUC metric with an end-to-end framework.

The ***major challenge*** of this task is that the objective function is not differentiable even with a proper surrogate loss function, suggesting that there is no easy way to perform end-to-end training. Facing this challenge, we propose a generic framework to approximately optimize the TPAUC with the help of deep learning. Generally speaking, our contributions are as follows.

First, we reformulate the original optimization problem as a bi-level optimization problem, where the inner-level problem provides a sparse sample selection process and the outer-level problem minimizes the loss over the selected instances.

On top of the reformulation, we propose a generic framework to construct surrogate optimization problems for the original problem. In the core of this framework lies the interplay of the surrogate penalty functions and surrogate weighting functions defined in this paper. Moreover, we construct a dual correspondence between these two classes of functions, such that we can easily find a standard single-level surrogate optimization problem whenever a surrogate penalty or a surrogate weighting function is obtained.

We then proceed to explore theoretical guarantees for the framework. On one hand, by comparing the surrogate problem and the original problem, we provide a mild sufficient condition under which the objective function surrogate problem becomes an upper bound of the original problem and further show that concave weighting function tends to be a better choice than their convex counterparts. On the other hand, we show that optimizing the surrogate problems could leverage reasonable generalization performance in terms of TPAUC with high probability.

2. Prior Art

Partial AUC Optimization. Comparing with existing studies to optimize partial AUC (Narasimhan & Agarwal, 2013b;c; 2017b), the key difference is two-fold. The previous studies only focus on a one-way partial AUC, where only the FPR is restricted within $[\alpha, \beta]$; while we are the first to study TPAUC optimization, a new AUC metric where both TPR and FPR are truncated. Moreover, most related studies are based on the cutting plane algorithm, which do not fit to the end-to-end training framework in deep learning. In our work, getting rid of complicated combinatorial optimization techniques, we propose a general framework to construct much simpler surrogate optimization problems for TPAUC that supports end-to-end training. ***Please see Appendix.A for a review of the general AUC optimization methods.***

3. Preliminaries

3.1. Standard AUC metric

Before showing the formal definition of the two-way partial AUC, we first provide a quick review of the standard AUC metric. Under the context of binary classification problems, an instance is denoted as $(\boldsymbol{x}, y)$, where $\boldsymbol{x} \in \mathcal{X}$ is the input raw features and $y \in \{0, 1\}$ is the label. Taking a step further, given a dataset $\mathcal{D}$, denote by $\mathcal{X}_P$ the set of positive instances in our dataset, and $\mathcal{X}_N$ the set of the negative ones, then the sampling process could be expressed as:

$$\mathcal{X}_P = \{\boldsymbol{x}_i^+\}_{i=1}^{n_+} \overset{i.i.d}{\sim} \mathcal{P}: \ \mathbb{P}\left[\boldsymbol{x}^+ | y = 1\right],$$
$$\mathcal{X}_N = \{\boldsymbol{x}_j^-\}_{j=1}^{n_-} \overset{i.i.d}{\sim} \mathcal{N}: \ \mathbb{P}\left[\boldsymbol{x}^- | y = 0\right],$$

where n_+, n_- are the numbers of positive/negative instances, respectively; and $\mathcal{P}, \mathcal{N}$ are the corresponding conditional distributions. For binary class problems, our goal is to learn a score function $f_{\boldsymbol{\theta}} : \mathcal{X} \to [0, 1]$, such that $f_{\boldsymbol{\theta}}(\boldsymbol{x})$ is proportional to the possibility that $\boldsymbol{x}$ belongs to the positive class. Based on the score function, we can further predict the label of an instance $\boldsymbol{x}$ as $\mathbf{1}\left[f_{\boldsymbol{\theta}}(\boldsymbol{x}) > t\right]$, where t is the decision threshold, $\mathbf{1}\left[\cdot\right]$ **is the indicator function**. Given a threshold t, we can define two elementary metrics known as True Positive Rate (TPR) and False Positive Rate (FPR), which are the probabilities that a positive/negative instance is predicted as a positive instance, i.e:

$$\begin{aligned} \mathsf{TPR}_{f_{\boldsymbol{\theta}}}(t) &= \mathop{\mathbb{P}}_{\boldsymbol{x}^+ \in \mathcal{P}}\left[f_{\boldsymbol{\theta}}(\boldsymbol{x}^+) > t\right], \\ \mathsf{FPR}_{f_{\boldsymbol{\theta}}}(t) &= \mathop{\mathbb{P}}_{\boldsymbol{x}^- \in \mathcal{N}}\left[f_{\boldsymbol{\theta}}(\boldsymbol{x}^-) > t\right]. \end{aligned} \tag{1}$$

Based on the label predictions, AUC is defined as the Area under the Receiver Operating Characteristic (ROC) curve plotted by True Positive Rate (TPR) against False Positive Rate (FPR) with varying thresholds, which could be expressed mathematically as follows:

$$\mathsf{AUC}(f_{\boldsymbol{\theta}}) = \int_0^1 \mathsf{TPR}_{f_{\boldsymbol{\theta}}}\left(\mathsf{FPR}_{f_{\boldsymbol{\theta}}}^{-1}(t)\right) dt. \tag{2}$$

When the possibility to observe a tied comparison is null, i.e.

$$\mathop{\mathbb{P}}_{\boldsymbol{x}^+ \in \mathcal{P}, \boldsymbol{x}^- \in \mathcal{N}}\left[f_{\boldsymbol{\theta}}(\boldsymbol{x}^+) = f_{\boldsymbol{\theta}}(\boldsymbol{x}^-)\right] = 0,$$

AUC is known (Hanley & McNeil, 1982) to enjoy a much simpler formulation as the possibility that correct ranking takes place between a positive and negative instance:

$$\mathsf{AUC}(f_{\boldsymbol{\theta}}) = 1 - \mathop{\mathbb{E}}_{\boldsymbol{x}^+ \sim \mathcal{P}}\left[\mathop{\mathbb{E}}_{\boldsymbol{x}^- \sim \mathcal{N}}\left[\ell_{0,1}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}^+) - f_{\boldsymbol{\theta}}(\boldsymbol{x}^-)\right)\right]\right],$$

where $\ell_{0,1}$ denotes the $0-1$ loss with $\ell_{0,1}(\boldsymbol{x}) = 1$ if $x < 0$, and $\ell_{0,1}(\boldsymbol{x}) = 0$, otherwise. Given a finite dataset $\mathcal{S} = \mathcal{X}_P \cup \mathcal{X}_N$, the unbiased estimation of $\mathsf{AUC}(f_{\boldsymbol{\theta}})$ could be expressed as:

$$\hat{\mathsf{AUC}}(f_{\boldsymbol{\theta}}) = 1 - \sum_{i=1}^{n_+}\sum_{j=1}^{n_-} \frac{\ell_{0,1}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^+) - f_{\boldsymbol{\theta}}(\boldsymbol{x}_j^-)\right)}{n_+ n_-}.$$

3.2. Two-Way Partial AUC Metrics

Definitions. As presented in the introduction, instead of the complete area of ROC, we focus on the area of ROC in a partial region with $\mathsf{TPR}_{f_{\boldsymbol{\theta}}}(t) \geq 1-\alpha, \mathsf{FPR}_{f_{\boldsymbol{\theta}}}(t) \leq \beta$, which is called two-way partial AUC in (Yang et al., 2019). Here we define it as $\mathsf{AUC}_\alpha^\beta$:

$$\begin{aligned} \mathsf{AUC}_\alpha^\beta(f_{\boldsymbol{\theta}}) = &\int_{\mathsf{FPR}_{f_{\boldsymbol{\theta}}}\left(\mathsf{TPR}_{f_{\boldsymbol{\theta}}}^{-1}(1-\alpha)\right)}^{\beta} \mathsf{TPR}_{f_{\boldsymbol{\theta}}}\left(\mathsf{FPR}_{f_{\boldsymbol{\theta}}}^{-1}(t)\right) dt \\ &- (1-\alpha)\cdot\left(\beta - \mathsf{FPR}_{f_{\boldsymbol{\theta}}}\left(\mathsf{TPR}_{f_{\boldsymbol{\theta}}}^{-1}(1-\alpha)\right)\right). \end{aligned}$$

Since the data distributions $\mathcal{P}, \mathcal{N}$ are often unknown, it is necessary to study its empirical estimation based on an observed dataset $\mathcal{S}$. (Yang et al., 2019) derives an empirical version of $\mathsf{AUC}_\alpha^\beta(f_{\boldsymbol{\theta}})$ as the truncated AUC over the hard positive and negative instances, which is denoted as $\hat{\mathsf{AUC}}_\alpha^\beta(f_{\boldsymbol{\theta}}, \mathcal{S})$ in our paper:

$$\hat{\mathsf{AUC}}_\alpha^\beta(f_{\boldsymbol{\theta}}, \mathcal{S}) = 1 - \sum_{i=1}^{n_+^\alpha}\sum_{j=1}^{n_-^\beta} \frac{\ell_{0,1}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{(i)}^+) - f(\boldsymbol{x}_{(j)}^-)\right)}{n_+ n_-}$$

where $\boldsymbol{x}_{(i)}^+$ denotes the hard positive instance that achieves ***bottom-i*** score among all positive instances, and $\boldsymbol{x}_{(j)}^-$ denotes the hard negative instance achieves ***top-j*** score among all negative instances, $n_+^\alpha = \lfloor n_+ \cdot \alpha \rfloor$, and $n_-^\beta = \lfloor n_- \cdot \beta \rfloor$, are the numbers of the chosen hard positive and negative examples. ***Please see Appendix.B for an analysis of the inconsistency between TPAUC and OPAUC.***

4. The Proposed Framework

4.1. A Generic Framework to Construct Surrogate Optimization Problems

Based on the empirical estimation shown in Sec.3.2, it is clear that optimizing TPAUC over a finite dataset $\mathcal{S}$ requires minimizing the following quantity:

$$1 - \hat{\mathsf{AUC}}_\alpha^\beta(f_{\boldsymbol{\theta}}, \mathcal{S}) = \sum_{i=1}^{n_+^\alpha}\sum_{j=1}^{n_-^\beta} \frac{\ell_{0,1}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{(i)}^+) - f(\boldsymbol{x}_{(j)}^-)\right)}{n_+ n_-}.$$

Following the framework of surrogate loss (Mohri et al., 2018), we replace the non-differential 0-1 loss with a convex loss function ℓ, such that $\ell(t)$ is an upper bound of $\ell_{0,1}(t)$. Note that if the scores live in $[0, 1]$, standard loss functions such as $\ell_{\exp}(t) = \exp(-t)$, $\ell_{sq}(t) = (1-t)^2$ often satisfy this constraint. Hence given a feasible surrogate loss ℓ, our goal is then to solve the following problem:

$$(OP_0)\ \min_{\boldsymbol{\theta}} \hat{\mathcal{R}}_{\alpha,\beta}^{\ell}(S, f_{\boldsymbol{\theta}}) = \sum_{i=1}^{n_+^\alpha}\sum_{j=1}^{n_-^\beta} \frac{\ell\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{(i)}^+) - f(\boldsymbol{x}_{(j)}^-)\right)}{n_+ n_-}.$$

Unfortunately, even with the choice of differentiable surrogate losses, the objective function $\hat{\mathcal{R}}_{\alpha,\beta}^{\ell}(S, f_{\boldsymbol{\theta}})$ is still not differentiable. This is because calculating $\boldsymbol{x}_{(i)}^+, \boldsymbol{x}_{(j)}^-$ requires sorting the scores of positive and negative instances. Nonetheless, the objective function is essentially a composition of a sparse sample selection operation and the original loss. This is shown in the following proposition, where (OP_0) is reformulated as a so-called bi-level optimization problem (Liu et al., 2020c;b). The inner-level problems provide a sparse sample selection process, and the outer-level problem performs the optimization based on the chosen instances. ***Please see Appendix.C for the proof.***

Proposition 1. *For any $\alpha, \beta \in (0,1)$, if scores $f_{\boldsymbol{\theta}}(\boldsymbol{x}) \in [0,1]$, and there are no ties in the scores, the original optimization problem is equivalent to the following problem:*

$$\min_{\boldsymbol{\theta}} \frac{1}{n_+ n_-} \sum_{i=1}^{n_+} \sum_{j=1}^{n_-} v_i^+ \cdot v_j^- \cdot \ell(f_{\boldsymbol{\theta}}, \boldsymbol{x}_i^+, \boldsymbol{x}_j^-)$$

$$s.t.\ v_+ = \operatorname*{argmax}_{v_i^+ \in [0,1], \sum_{i=1}^{n_+} v_i^+ \le n_+^\alpha} \sum_{i=1}^{n_+} \left(v_i^+ \cdot (1 - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^+))\right)$$

$$v_- = \operatorname*{argmax}_{v_j^- \in [0,1], \sum_{j=1}^{n_-} v_j^- \le n_-^\beta} \sum_{j=1}^{n_-} \left(v_j^- \cdot f_{\boldsymbol{\theta}}(\boldsymbol{x}_j^-)\right)$$

where

$$\ell(f_{\boldsymbol{\theta}}, \boldsymbol{x}_i^+, \boldsymbol{x}_j^-) = \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^+) - f_{\boldsymbol{\theta}}(\boldsymbol{x}_j^-)).$$

Based on the proposition, the source of the intractability of (OP_0) comes from the ℓ_1 ball constraints $\sum_{i=1}^{n_+} v_i^+ \le n_+^\alpha$, $\sum_{j=1}^{n_-} v_j^- \le n_-^\beta$ in the inner-level problem. To establish an efficient approximation of the original problem, we follow a standard trick to transform the ℓ_1 ball constraints to ℓ_1 penalty terms in the objective function (note that v_+, v_- are non-negative). In this way, the inner-level problems become:

$$v_+ = \operatorname*{argmax}_{v_i^+ \in [0,1]} \sum_{i=1}^{n_+} \left(v_i^+ \cdot (1 - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^+)) - \lambda^+ \cdot v_i^+\right)$$

$$v_- = \operatorname*{argmax}_{v_j^- \in [0,1]} \sum_{j=1}^{n_-} \left(v_j^- \cdot f_{\boldsymbol{\theta}}(\boldsymbol{x}_j^-) - \lambda^- \cdot v_j^-\right)$$

Furthermore, to avoid sparisty, we replace the sparsity-inducing ℓ_1 penalty with a smooth surrogate φ_γ. This naturally leads to a smooth problem:

$$(OP_1)\ \min_{\boldsymbol{\theta}} \frac{1}{n_+^\alpha n_-^\beta} \sum_{i=1}^{n_+} \sum_{j=1}^{n_-} v_i^+ \cdot v_j^- \cdot \ell(f_{\boldsymbol{\theta}}, \boldsymbol{x}_i^+, \boldsymbol{x}_j^-)$$

$$s.t\ \ v_+ = \operatorname*{argmax}_{v_i^+ \in [0,1]} \sum_{i=1}^{n^+} \left(v_i^+ \cdot (1 - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^+)) - \varphi_\gamma(v_i^+)\right)$$

$$v_- = \operatorname*{argmax}_{v_j^- \in [0,1]} \sum_{j=1}^{n_-} \left(v_j^- \cdot f_{\boldsymbol{\theta}}(\boldsymbol{x}_j^-) - \varphi_\gamma(v_j^-)\right)$$

To ensure that the chosen φ_γ provides an effective approximation of the ℓ_1 penalty, we pose several regularities on such functions. In the following, we define this class of functions as the calibrated smooth penalty function.

Definition 1. *A penalty function $\varphi_\gamma(x) : \mathbb{R}_+ \to \mathbb{R}$ is called a* **calibrated smooth penalty function**, *if it satisfies the following regularities:*

(A) *φ_γ has continuous third-order derivatives.*

(B) *φ_γ is strictly increasing in the sense that $\varphi'_\gamma(x) > 0$.*

(C) *φ_γ is strictly convex in the sense that $\varphi''_\gamma(x) > 0$.*

(D) *φ_γ has positive third-order derivatives in the sense that $\varphi'''_\gamma(x) > 0$.*

Note that the condition (B) is inherited from the ℓ_1 norm. While the other conditions improve the smoothness of the function. Moreover, the last condition is to ensure that the weighting function is strictly concave (see the arguments about the weights).

Given the penalty functions, we turn to explore a corresponding factor in the framework. According to the inner level problem, the sample weights v_i^+, v_j^- have a dual correspondence with the penalty functions. More precisely, given a fixed ϕ_γ, one can derive the corresponding weighting function ψ_γ as a function of $f_{\boldsymbol{\theta}}(\boldsymbol{x})$ such that:

$$v_i^+ = \psi_\gamma(1 - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^+)),\ v_j^- = \psi_\gamma(f_{\boldsymbol{\theta}}(\boldsymbol{x}_j^-)),\ v_i^+, v_j^- \in [0,1].$$

Moreover, if ψ_γ has a closed-form expression, then we can cancel the inner optimization problem and instead minimize the following weighted empirical risk $\hat{\mathcal{R}}_\psi^\ell$:

$$\hat{\mathcal{R}}_\psi^\ell(\mathcal{S}, f_{\boldsymbol{\theta}}) = \frac{1}{n_+ n_-} \sum_{i=1}^{n_+} \sum_{j=1}^{n_-} \psi_\gamma(1 - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^+)) \cdot \psi_\gamma(f_{\boldsymbol{\theta}}(\boldsymbol{x}_j^-)) \cdot \ell(f_{\boldsymbol{\theta}}, \boldsymbol{x}_i^+, \boldsymbol{x}_j^-). \tag{3}$$

In this sense, adopting a smooth penalty function ends up with a dual soft weighting strategy over the hard instances. Again, to reach a proper weight function, we also require it to satisfy some necessary regularities. In the following, we define this class of functions as the calibrated weighting function.

Definition 2. *A weighting function $\psi_\gamma(x) : [0,1] \to \mathsf{Rng}$, where $\mathsf{Rng} \subseteq [0,1]$, is called a* **calibrated weighting function**, *if it satisfies the following regularities:*

(A) *ψ_γ has continuous second-order derivatives.*

(B) *ψ_γ is strictly increasing in the sense that $\psi'_\gamma(x) > 0$.*

(C) *ψ_γ is strictly concave in the sense that $\psi''_\gamma(x) < 0$.*

In this definition, (B) is a natural requirement to make the weight proportional to the target instance's difficulty. Condition (C) is an interesting trait in our framework. To see why this is necessary, let us note that the weight functions v_i^+, v_j^- are continuous surrogates residing in $[0,1]$ for threshold

function

$$\mathbf{1}\left[1 - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^+) > 1 - f_{\boldsymbol{\theta}}(\boldsymbol{x}_{(n_+^\alpha)}^+)\right],$$
$$\mathbf{1}\left[f_{\boldsymbol{\theta}}(\boldsymbol{x}_j^-) > f_{\boldsymbol{\theta}}(\boldsymbol{x}_{(n_-^\beta)}^-)\right],$$

respectively. To be simple, we continue our discussion with a general form $\mathbf{1}[x > 0]$. Obviously, weight decay for large x should be smooth such that the loss could attend at the top $f_{\boldsymbol{\theta}}(\boldsymbol{x}^+)$ and $(1 - f_{\boldsymbol{\theta}}(\boldsymbol{x}^-))$ scores. Moreover, to avoid overfitting, the model should as well have sufficient memory of the easy examples. Hence the weights for such examples should not be too close to zero. These observations are exactly typical traits for a concave function. As shown in Fig.2, we visualize the difference between a convex function $y = x^2$ and $y = \mathbf{1}[x > 0.5]$, and the difference between concave function $y = x^{0.05}$ and $y = \mathbf{1}[x > 0.5]$.

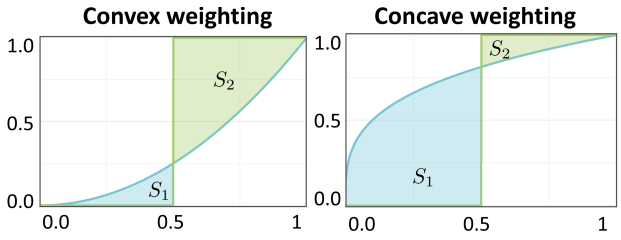


Figure 2. convex vs. concave weighting functions.

Another reason to choose concave functions is that they can benefit the optimization process. More precisely, we expect the loss function $\hat{\mathcal{R}}_\psi^\ell$ in Eq. (3) to be an upper bound of $\hat{\mathcal{R}}_{\alpha,\beta}^\ell$, such that minimizing $\hat{\mathcal{R}}_\psi^\ell$ could also minimize the original loss. Back to Fig.2, this is more likely to happen if S_1/S_2 is large. In fact, this is a condition which is much easier for concave functions to satisfy. From a quantitative perspective, the following proposition provides a sufficient and a necessary condition under which $\hat{\mathcal{R}}_\psi^\ell \geq \hat{\mathcal{R}}_{\alpha,\beta}^\ell$. Moreover, it shows that it is generally more challenging for a convex function to realize an upper bound. ***Please see Appendix.D for the proof.***

Proposition 2. *Given a strictly increasing weighting function $\psi_\gamma : [0,1] \to [0,1]$, such that $v_i^+ = \psi_\gamma(1 - f_{\boldsymbol{\theta}}(\boldsymbol{x}_i^+))$, $v_j^- = \psi_\gamma(f_{\boldsymbol{\theta}}(\boldsymbol{x}_j^-))$, denote:*

$$\mathcal{I}_1^+ = \left\{x_+ : x_+ \in \mathcal{X}_P, f(x_+) \geq f(x_+^{(n_+^\alpha)})\right\},$$
$$\mathcal{I}_1^- = \left\{x_- : x_- \in \mathcal{X}_N, f(x_-) \leq f(x_-^{(n_-^\beta)})\right\},$$

denote $\mathcal{I}_2$ as $(\mathcal{X}_P \times \mathcal{X}_N)\backslash(\mathcal{I}_1^+ \times \mathcal{I}_1^-)$; denote $\bar{\mathbb{E}}_{\boldsymbol{x}^+ \in \mathcal{I}_1^+}[x]$ as the empirical expectation of x over the set $\mathcal{I}_1^+$, and $\bar{\mathbb{E}}_{\boldsymbol{x}^- \in \mathcal{I}_1^-}[x]$, $\bar{\mathbb{E}}_{\boldsymbol{x}^+ \in \mathcal{I}_1^+, \boldsymbol{x}^- \in \mathcal{I}_1^-}$, $\bar{\mathbb{E}}_{\boldsymbol{x}^+, \boldsymbol{x}^- \in \mathcal{I}_2}$ are defined similarly; define $l_{i,j} = \ell(f_{\boldsymbol{\theta}}, \boldsymbol{x}_i^+, \boldsymbol{x}_j^-)$. We assume that

$$n_+^\alpha \in \mathbb{N},\ n_-^\beta \in \mathbb{N},\ f_{\boldsymbol{\theta}}(\boldsymbol{x}^+), f_{\boldsymbol{\theta}}(\boldsymbol{x}^-) \in (0,1),$$

then:

(a) A sufficient condition for $\hat{\mathcal{R}}_{\alpha,\beta}^\ell(\mathcal{S}, f_{\boldsymbol{\theta}}) \leq \hat{\mathcal{R}}_\psi^\ell(\mathcal{S}, f_{\boldsymbol{\theta}})$ is that:

$$\sup_{p \in (0,1), q = -\frac{p}{1-p}} [\rho_p - \xi_q] \geq 0,$$

where

$$\rho_p = \frac{\left(\bar{\mathbb{E}}_{\boldsymbol{x}^+, \boldsymbol{x}^- \in \mathcal{I}_2}\left[v_+^p \cdot v_-^p\right]\right)^{1/p}}{\left(\bar{\mathbb{E}}_{\boldsymbol{x}^+ \in \mathcal{I}_1^+, \boldsymbol{x}^- \in \mathcal{I}_1^-}\left[(1 - v_+ v_-)^2\right]\right)^{1/2}},$$
$$\xi_q = \frac{\alpha\beta}{1-\alpha\beta} \cdot \frac{\left(\bar{\mathbb{E}}_{\boldsymbol{x}^+, \boldsymbol{x}^- \in \mathcal{I}_2}(\ell_{i,j}^2)\right)^{1/2}}{\left(\bar{\mathbb{E}}_{\boldsymbol{x}^+ \in \mathcal{I}_1^+, \boldsymbol{x}^- \in \mathcal{I}_1^-}(\ell_{i,j}^q)\right)^{1/q}}.$$

(b) If there exists at least one strictly concave ψ_γ such that the $\hat{\mathcal{R}}_{\alpha,\beta}^\ell(\mathcal{S}, f_{\boldsymbol{\theta}}) > \hat{\mathcal{R}}_\psi^\ell(\mathcal{S}, f_{\boldsymbol{\theta}})$, then $\hat{\mathcal{R}}_{\alpha,\beta}^\ell(\mathcal{S}, f_{\boldsymbol{\theta}}) > \hat{\mathcal{R}}_\psi^\ell(\mathcal{S}, f_{\boldsymbol{\theta}})$ holds for all convex ψ_γ.

According to Prop.2, $\hat{\mathcal{R}}_\psi^\ell$ can achieve the upper bound of $\hat{\mathcal{R}}_{\alpha,\beta}^\ell$ if α, β are small, and the empirical distribution has significant masses at instances with moderate difficulty.

Dual Correspondence Theory. Now with the penalty function and weighting function clarified, we establish their dual correspondence with the following proposition. ***Please see Appendix.E for the proof.***

Proposition 3. *Given a strictly convex function φ_γ, and define $\psi_\gamma(t)$ as:*

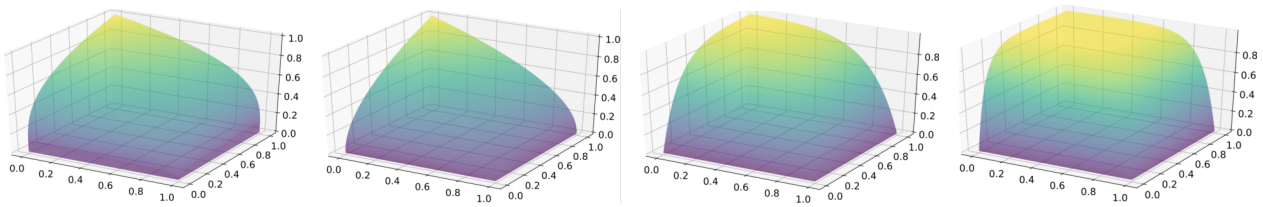
$$\psi_\gamma(t) = \operatorname*{argmax}_{v \in [0,1]} v \cdot t - \varphi_\gamma(v),$$

then we can draw the following conclusions:

(a) If φ_γ is a calibrated smooth penalty function, we have $\psi_\gamma(t) = \varphi_\gamma'^{-1}(t)$, which is a calibrated weighting function.

(b) If ψ_γ is a calibrated weighting function such that $v = \psi_\gamma(t)$, we have $\varphi_\gamma(v) = \int \psi_\gamma^{-1}(v)dv + const.$, which is a calibrated smooth penalty function.

According to Prop.3, given a calibrated smooth penalty function, one can obtain an implicit soft weighting strategy via Prop.3-(a). Likewise, given a calibrated weighting function, one can find an implicit regularizer over the sample weights via Prop.3-(b). Based on the regularities of the two



(a) $(\gamma-1)^{-1}=0.3$ (b) $(\gamma-1)^{-1}=0.5$ (c) $\gamma=5$ (d) $\gamma=10$

Figure 3. Visualization of the Landscape of the pairwise weights $\psi_\gamma(x)\cdot\psi_\gamma(y)$. Here, (a) and (b) plot $\psi_\gamma^{\mathsf{poly}}$, while (c) and (d) plot $\psi_\gamma^{\mathsf{exp}}$.

components, both ψ and φ have closed-form formulation if the other one is known. This means that we can solve the bi-level optimization framework in (OP_1) by simply minimizing the resulting $\hat{\mathcal{R}}^{\ell}_{\psi}(\mathcal{S}, f_{\boldsymbol{\theta}})$, leading to a much simpler optimization that can be solved directly by an end-to-end training framework. *Consequently, the dual correspondence theory provides a simple way to establish a surrogate optimization problem of TPAUC, once a weighting function or a penalty function is at hand.*

4.2. Two Instantiations of the Generic Framework

Based on the generic framework, in this subsection, we provide two practical instantiations.

Polynomial Surrogate Model. From the penalty function perspective, the original ℓ_1 penalty realizes $\psi_\gamma=\gamma\cdot t$. In this way, it is a natural choice to adopt a polynomial penalty $\varphi_\gamma^{\mathsf{poly}}(t)=C\cdot t^\gamma$ as a dense surrogate for ℓ_1. Inspired by this, we propose a polynomial surrogate model as example 1.

Example 1 (Polynomial Surrogate Model)**.** *In the polynomial surrogate model, we set:*

$$\varphi_\gamma^{\mathsf{poly}}(t)=\frac{1}{\gamma}\cdot t^\gamma,\ \psi_\gamma^{\mathsf{poly}}(t)=t^{\frac{1}{\gamma-1}},\ \gamma>2$$

The visualizations of the weights are shown in Fig.3.

Example 2 (Exponential Surrogate Model)**.** *In the exponential surrogate model, we set:*

$$\varphi_\gamma^{\mathsf{exp}}(t)=\frac{(1-t)(\log(1-t)-1)+1}{\gamma},\ \psi_\gamma^{\mathsf{exp}}(t)=1-e^{-\gamma t}$$

Exponential Surrogate Model. Considering the properties of the weighting functions, we expect that ψ_γ will have a flat landscape for large t. Motivated by this, we adopt an exponential weighting function $\psi_\gamma^{\mathsf{exp}}(t)=1-e^{-\gamma t}$ (the landscape is shown in Fig.3 (c)-(d)). The resulting model is then shown as Exp.2. The visualizations of the weights are shown in Fig.3.

4.3. Generalization Analysis

In this subsection, we turn to explore how generalization error behaves away from the training error in terms of the TPAUC metric. In other words, we will show when a well-trained model will lead to a reasonable generalization performance. Our analysis is based on a standard assumption that the classifiers are chosen from a hypothesis class $\mathcal{F}$ (e.g. the class of a specific type of deep neural networks). The key challenge here is that $\hat{\mathcal{R}}^{\ell}_{\alpha,\beta}(\mathcal{S}, f_{\boldsymbol{\theta}})$ is not an unbiased estimation of $\mathsf{AUC}^{\beta}_{\alpha}(f_{\boldsymbol{\theta}},\mathcal{S})$, making standard generalization analysis (Mohri et al., 2018) unavailable. Here we extend the error decomposition technique for OPAUC (Narasimhan & Agarwal, 2017b) and employ the result in Prop.2 to reach the following theorem. ***Please see Appendix.F for the proof.***

Theorem 1 (***Informal***)**.** *Assume that there are no ties in the datasets, and the surrogate loss function ℓ with range $[0,1]$, is an upper bound of the 0-1 loss, then, for all $f_{\boldsymbol{\theta}}\in\mathcal{F}$, and all $\alpha,\beta\in(0,1)$ such that condition (a) of Proposition 2 holds, the following inequality holds with high probability:*

$$\mathcal{R}^{\alpha,\beta}_{AUC}(f_{\boldsymbol{\theta}},\mathcal{S})\leq\hat{\mathcal{R}}^{\ell}_{\psi}(f_{\boldsymbol{\theta}},\mathcal{S})+\tilde{\mathcal{O}}\left((\frac{\mathsf{VC}}{n_+})^{1/2}+(\frac{\mathsf{VC}}{n_-})^{1/2}\right),$$

where $\tilde{\mathcal{O}}$ is the big-O complexity notation hiding the logarithm factors, $\mathcal{R}^{\alpha,\beta}_{AUC}(f_{\boldsymbol{\theta}},\mathcal{S})=1-\mathsf{AUC}^{\beta}_{\alpha}(f_{\boldsymbol{\theta}},\mathcal{S})$, and VC is the VC dimension of the hypothesis class:

$$\mathcal{T}(\mathcal{F})\triangleq\{\mathsf{sign}(f_{\boldsymbol{\theta}}(\cdot)-\delta):f_{\boldsymbol{\theta}}\in\mathcal{F},\ \delta\in\mathbb{R}\}.$$

According to the theorem, for all α,β satisfying condition (a) of Prop.2 and any model in $\mathcal{F}$, the generalization error represented by the loss version of TPAUC $\mathcal{R}^{\alpha,\beta}_{AUC}(f_{\boldsymbol{\theta}},\mathcal{S})=1-\mathsf{AUC}^{\beta}_{\alpha}(f_{\boldsymbol{\theta}},\mathcal{S})$ is no larger than the empirical loss $\hat{\mathcal{R}}^{\ell}_{\psi}(f_{\boldsymbol{\theta}},\mathcal{S})$ plus a complexity term. The complexity term is affected by two factors. On one hand, it vanishes with large enough training datasets. On the other hand, it remains moderate if the model hypothesis class's VC dimension is not too large. Moreover, moderate upper bounds for the VC dimension are now available for typical models ranging from linear models to deep neural

networks. Finally, for a well-trained model, the empirical loss $\hat{\mathcal{R}}^{\ell}_{\psi}(f_{\boldsymbol{\theta}}, \mathcal{S})$ is restricted to be small in our framework; one can then reach reasonable generalization results with high probability.

5. Experiments

In this section, we present our empirical results and some of the details of the experiments. ***Please see Appendix.G for more details on the settings and results.***

5.1. Competitors

To validate the effectiveness of our proposed methods, we consider two types of competitors in our experiments. On one hand, we compare our proposed methods with other methods dealing with imbalanced data. The competitors include class-reweighted **CE**, **Focal** (Lin et al., 2017), **CB-CE** (Cui et al., 2019), and **CE-Focal** (Cui et al., 2019). On the other hand, we also include a standard AUC optimization method as our baseline. Here we use the square function $\ell_{sq}(t) = (1-t)^2$ as the surrogate loss, which is widely-adopted in AUC optimization studies. The resulting competitor is named **SqAUC**. Finally, we implement our polynomial surrogate model and the exponential surrogate model on top of **SqAUC**, which are denoted as **Poly** and **Exp** in the rest of this section.

5.2. Evaluation Metrics

Aiming at optimizing the TPAUC metrics, we consider TPAUC with $\alpha = 0.3, \beta = 0.3$, $\alpha = 0.4, \beta = 0.4$, $\alpha = 0.5, \beta = 0.5$, respectively. Moreover, to normalize the range of their magnitude to $[0, 1]$, we adopt the following variant of the TPAUC metric:

$$\mathsf{TPAUC}(\alpha, \beta) = 1 - \sum_{i=1}^{n_+^{\alpha}} \sum_{j=1}^{n_-^{\beta}} \frac{\ell_{0,1}\left(f_{\boldsymbol{\theta}}(\boldsymbol{x}_{(i)}^{+}) - f(\boldsymbol{x}_{(j)}^{-})\right)}{n_+^{\alpha} n_-^{\beta}}.$$

5.3. Dataset Description

Note that AUC is aimed at dealing with binary classification problems, hence we construct long-tail binary datasets as follows.

Binary CIFAR-10-LT Dataset. We create a long-tailed CIFAR-10 dataset, where the sample sizes across different classes decay exponentially, and the ratio of sample sizes of the least frequent to the most frequent class ρ is set to 0.01. Afterwards, we create 3 binary long-tailed datasets based on CIFAR-10-LT by selecting one category as positive examples and the others as negative examples.

Binary CIFAR-100-LT Dataset. We create 3 CIFAR-100-LT subsets in the same way as CIFAR-10-LT, where a superclass is selected as positive examples each time.

Binary Tiny-ImageNet-200-LT Dataset. The original Tiny-ImageNet-200 dataset contains 100,000 colour images sourced from 200 different categories, with 500 images for each category. Similar to the CIFAR-100-LT dataset, we choose 3 positive superclasses to construct binary subsets.

5.4. Warm-Up Training Phase With Delay Epochs

Focusing on the hard examples at the beginning of the training process brings a high risk of over-fitting. It is thus necessary to focus on the entire dataset to capture the global information. Inspired by this investigation, we adopt a warm-up training strategy. Specifically, the model will go through a warm-up phase with E_k **Epochs** of ordinary AUC optimization training. Afterward, we start the TPAUC training phase by optimizing our proposed surrogate problems. We will show its effect in the next subsection.

5.5. Overall Performance

The performances of all the involved methods on three subsets of CIFAR-10-LT, CIFAR-100-LT, and Tiny-Imagent-200-LT are shown in Tab.1. For each method here, the results for different TPAUC metrics are tuned independently. Consequently, we have the following observations: 1) The best performance of our proposed methods consistently surpasses all the competitors significantly on all the datasets, except the result for $\mathsf{TPAUC}(0.4, 0.4)$ and $\mathsf{TPAUC}(0.5, 0.5)$ on subset 2 of Tiny-Imagent-200-LT and $\mathsf{TPAUC}(0.3, 0.3)$ for subset 3 of Tiny-Imagent-200-LT. Our proposed methods attain fairly competitive results compared with the competitors even for the three failure results. 2) The performance improvement is especially sharp on $\mathsf{TPAUC}(0.3, 0.3)$. This suggests the ability of our proposed methods to optimize the head region under the ROC curve.

5.6. Sensitivity Analysis

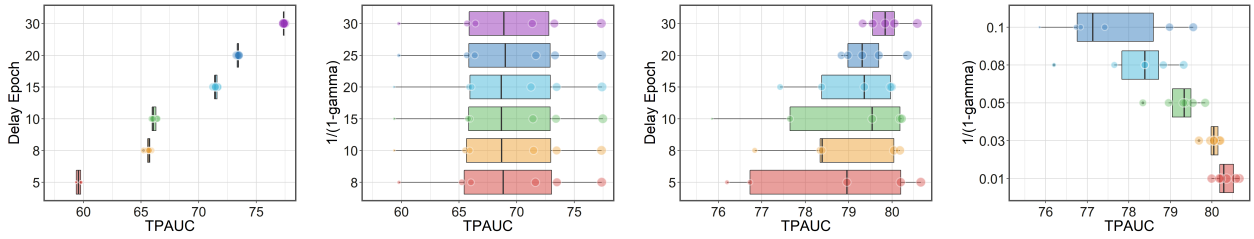
Our proposed framework evolves two hyperparameters: γ for loss functions and E_k for the warm-up strategy. Next, we analyze their effect respectively. Our analysis is based on a 2d grid search over E_k, γ. When investigating the effect of E_k (γ resp.), we will show the performance variation in terms of γ (E_k resp.) given each fixed E_k (γ resp.)

Effect of E_k. In Fig.4-(a), (c), we show the sensitivity in terms of E_k on subset 2 of CIFAR-10-LT for **Exp** and **Poly**, respectively. For **Exp**, we see that increasing E_k from 5 to 30 leads to a significantly increasing trend of performance. This shows that a warm-up phase is necessary for **Exp**. For **Poly**, we observe that the increasing trend of average performance is much weaker. This is probably because γ has a strong influence on the performance so that the variances become much larger in general.

Effect of γ. In Fig.4-(b), (d), we show the sensitivity in

Table 1. Performance Comparisons over different metrics and datasets, where (x, y) stands for $\mathsf{TPAUC}(x, y)$ in short.

dataset	type	methods	Subset1			Subset2			Subset3		
			(0.3,0.3)	(0.4,0.4)	(0.5,0.5)	(0.3,0.3)	(0.4,0.4)	(0.5,0.5)	(0.3,0.3)	(0.4,0.4)	(0.5,0.5)
CIFAR-10-LT	Competitors	CE-RW	9.09	30.86	47.99	72.83	83.33	88.71	23.47	44.44	59.69
		Focal	9.84	30.89	50.83	75.72	85.10	90.06	21.47	45.88	59.09
		CBCE	3.29	27.30	43.95	69.48	80.80	86.87	12.94	34.06	51.09
		CBFocal	9.04	31.73	48.13	77.99	86.75	91.13	21.32	43.03	59.11
		SqAUC	18.05	40.74	57.94	80.09	87.78	91.87	31.52	50.00	64.42
	Ours	Poly	**21.43**	**44.41**	**59.10**	80.66	**88.07**	**92.15**	**36.54**	**54.48**	67.19
		Exp	20.86	41.78	58.38	**81.22**	87.88	91.93	32.47	53.86	**67.32**
CIFAR-100-LT	Competitors	CE-RW	31.43	52.60	66.21	79.70	88.06	92.64	3.09	21.32	40.75
		Focal	36.51	61.71	73.25	83.08	90.35	93.76	8.09	28.88	49.89
		CBCE	17.53	38.79	55.19	67.91	79.32	85.82	1.84	18.46	37.04
		CBFocal	41.85	62.41	73.13	82.75	89.57	92.89	7.10	29.12	44.84
		SqAUC	63.24	76.62	84.68	91.02	93.69	94.73	41.60	60.36	70.86
	Ours-TPAUC	Poly	**68.02**	**79.11**	**85.17**	**91.13**	**93.78**	**95.69**	**47.07**	**65.89**	**75.08**
		Exp	63.24	77.94	84.62	90.69	93.74	95.41	44.54	64.58	73.02
Tiny-ImageNet-200-LT	Competitors	CE-RW	80.90	87.76	91.54	93.30	96.15	97.53	90.37	94.34	96.75
		Focal	81.18	88.06	91.72	93.23	96.08	97.59	91.35	94.87	96.63
		CBCE	80.64	87.58	91.17	93.77	**96.52**	**97.77**	91.66	95.19	96.79
		CBFocal	80.44	87.95	91.91	93.46	96.43	97.64	91.06	94.82	96.62
		SqAUC	80.16	87.99	91.67	93.10	96.07	97.32	**92.15**	95.16	96.75
	Ours-TPAUC	Poly	80.44	88.21	91.98	93.00	95.61	97.47	92.02	**95.25**	**96.84**
		Exp	**82.61**	**89.13**	**92.62**	**93.82**	96.12	97.38	91.25	94.78	96.57



(a) Effect of E_k on **Exp** (b) Effect of γ on **Exp** (c) Effect of E_k on **Poly** (d) Effect of γ on **Poly**

Figure 4. Sensitivity analysis on subset 2 of CIFAR-10-LT where TPAUC is measured with $\alpha = 0.3, \beta = 0.3$. For each box in (a) and (c), E_k is fixed as the y-axis value, and the scattered points along the box show the variation of γ. For each Box in (b) and (d), $(\gamma - 1)^{-1}$ is fixed as the y-axis value, and the scattered points along the box show the variation of E_k.

terms of γ on subset 2 of CIFAR-10-LT for **Exp** and **Poly**, respectively. One can observe very different trends on these two methods. This is because that **Exp** and **Poly** have different characteristics in terms of the landscape of the weight function. As shown in Fig.3-(c), (d), the weight landscape of **Exp** is flat within a large subset of the domain. In this sense, it does not have a strong dependency on γ. As shown in Fig.3-(a), (b), the weight landscape of **Poly** is more sensitive toward γ.

6. Conclusion

In this paper, we initiate the study on TPAUC optimization. Since the original optimization problem could not be solved directly with an end-to-end framework, we propose a general framework to construct surrogate optimization problems for TPAUC. Following our dual correspondence theory, we can establish a surrogate problem once a calibrated penalty function or a calibrated weighting function is found. To see how and when our framework could provide efficient approximations of the original problem, we show that the surrogate objective function could reach the upper bound of the original one and that concave weighting functions are better choices than their convex counterparts. Moreover, we also provide high probability uniform upper bounds for the generalization error. The experiments on three datasets consistently show the advantage of our framework.

7. Acknowledgement

This work was supported in part by the National Key R&D Program of China under Grant 2018AAA0102003, in part by National Natural Science Foundation of China: 61931008, 61620106009, 61836002, U2001202, U1736219, and 61976202, in part by Youth Innovation Promotion Association CAS, and in part by the Strategic Priority Re-

search Program of Chinese Academy of Sciences, Grant No. XDB28000000.

References

Agarwal, S. Surrogate regret bounds for bipartite ranking via strongly proper losses. *Journal of Machine Learning Research*, 15(1):1653–1674, 2014.

Agarwal, S., Graepel, T., Herbrich, R., Har-Peled, S., and Roth, D. Generalization bounds for the area under the roc curve. *Journal of Machine Learning Research*, 6(Apr): 393–425, 2005.

Alan, A. H. and Raskutti, B. Optimising area under the roc curve using gradient descent. *International Conference on Machine Learning*, pp. 49–56, 2004.

Calders, T. and Jaroszewicz, S. Efficient auc optimization for classification. *European Conference on Principles of Data Mining and Knowledge Discovery*, pp. 42–53, 2007.

Clémençon, S., Lugosi, G., Vayatis, N., et al. Ranking and empirical minimization of u-statistics. *The Annals of Statistics*, 36(2):844–874, 2008.

Cortes, C. and Mohri, M. Auc optimization vs. error rate minimization. *Advances in Neural Information Processing Systems*, pp. 313–320, 2003.

Cui, Y., Jia, M., Lin, T.-Y., Song, Y., and Belongie, S. Class-balanced loss based on effective number of samples. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9268–9277, 2019.

Fawcett, T. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

Freund, Y., Iyer, R., Schapire, R. E., and Singer, Y. An efficient boosting algorithm for combining preferences. *Journal of machine learning research*, 4(Nov):933–969, 2003.

Gao, W. and Zhou, Z. On the consistency of AUC pairwise optimization. *International Joint Conference on Artificial Intelligence*, pp. 939–945, 2015.

Gao, W., Wang, L., Jin, R., Zhu, S., and Zhou, Z. One-pass auc optimization. *International Conference on Machine Learning*, pp. 906–914, 2013.

Hand, D. J. and Till, R. J. A simple generalisation of the area under the roc curve for multiple class classification problems. *Machine Learning*, 45(2):171–186, 2001.

Hanley, J. A. and McNeil, B. J. The meaning and use of the area under a receiver operating characteristic (roc) curve. *Radiology*, 143(1):29–36, 1982.

Hao, H., Fu, H., Xu, Y., Yang, J., Li, F., Zhang, X., Liu, J., and Zhao, Y. Open-narrow-synechiae anterior chamber angle classification in AS-OCT sequences. In *International Conference on Medical Image Computing and Computer Assisted Intervention*, 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015.

Joachims, T. A support vector method for multivariate performance measures. *International Conference on Machine Learning*, pp. 377–384, 2005.

Joachims, T. Training linear svms in linear time. *the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 217–226, 2006.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. Focal loss for dense object detection. In *IEEE international conference on computer vision*, pp. 2980–2988, 2017.

Liu, C., Zhong, Q., Ao, X., Sun, L., Lin, W., Feng, J., He, Q., and Tang, J. Fraud transactions detection via behavior tree with local intention calibration. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2020a.

Liu, R., Li, Z., Zhang, Y., Fan, X., and Luo, Z. Bi-level probabilistic feature learning for deformable image registration. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 2020b.

Liu, R., Mu, P., Yuan, X., Zeng, S., and Zhang, J. A generic first-order algorithmic framework for bi-level programming beyond lower-level singleton. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, 2020c.

Liu, W., Luo, W., Lian, D., and Gao, S. Future frame prediction for anomaly detection - A new baseline. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6536–6545, 2018.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. Foundations of machine learning. 2018.

Narasimhan, H. and Agarwal, S. A structural SVM based approach for optimizing partial AUC. *International Conference on Machine Learning*, pp. 516–524, 2013a.

Narasimhan, H. and Agarwal, S. Svmpauctight: a new support vector method for optimizing partial auc based on a tight convex upper bound. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 167–175, 2013b.

Narasimhan, H. and Agarwal, S. A structural svm based approach for optimizing partial auc. In *International Conference on Machine Learning*, pp. 516–524, 2013c.

Narasimhan, H. and Agarwal, S. Support vector algorithms for optimizing the partial area under the ROC curve. *Neural Computation*, 29(7):1919–1963, 2017a.

Narasimhan, H. and Agarwal, S. Support vector algorithms for optimizing the partial area under the roc curve. *Neural computation*, 29(7):1919–1963, 2017b.

Natole, M., Ying, Y., and Lyu, S. Stochastic proximal algorithms for auc maximization. *International Conference on Machine Learning*, pp. 3707–3716, 2018.

Natole, M. A., Ying, Y., and Lyu, S. Stochastic auc optimization algorithms with linear convergence. *Frontiers in Applied Mathematics and Statistics*, 5:30, 2019.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. pp. 8024–8035, 2019.

Ralaivola, L., Szafranski, M., and Stempfel, G. Chromatic pac-bayes bounds for non-iid data: Applications to ranking and stationary β-mixing processes. *Journal of Machine Learning Research*, 11(Jul):1927–1956, 2010.

Sutskever, I., Martens, J., Dahl, G. E., and Hinton, G. E. On the importance of initialization and momentum in deep learning. In *ICML 2013*, pp. 1139–1147, 2013.

Usunier, N., Amini, M.-R., and Gallinari, P. A data-dependent generalisation error bound for the auc. *ICML Workshop on ROC Analysis in Machine Learning*, 2005.

Usunier, N., Amini, M. R., and Gallinari, P. Generalization error bounds for classifiers trained with interdependent data. *Advances in Neural Information Processing Systems*, pp. 1369–1376, 2006.

Wang, D., Lin, J., Cui, P., Jia, Q., Wang, Z., Fang, Y., Yu, Q., Zhou, J., Yang, S., and Qi, Y. A semi-supervised graph attentive network for financial fraud detection. In *2019 IEEE International Conference on Data Mining (ICDM)*, 2019.

Wu, H., Hu, Z., Jia, J., Bu, Y., He, X., and Chua, T.-S. Mining unfollow behavior in large-scale online social networks via spatial-temporal interaction. In *AAAI Conference on Artificial Intelligence*, volume 34, pp. 254–261, 2020.

Yang, H., Lu, K., Lyu, X., and Hu, F. Two-way partial auc and its properties. *Statistical Methods in Medical Research*, 28(1):184–195, 2019.

Ying, Y., Wen, L., and Lyu, S. Stochastic online auc maximization. *Advances in Neural Information Processing Systems*, pp. 451–459, 2016.

Zhang, X., Saha, A., and Vishwanathan, S. Smoothing multivariate performance measures. *Journal of Machine Learning Research*, 13(1):3623–3680, 2012.

Zhao, P., Hoi, S. C., Jin, R., and Yang, T. Online auc maximization. In *International Conference on Machine Learning*, pp. 233–240, 2011.

Zhou, K., Gao, S., Cheng, J., Gu, Z., Fu, H., Tu, Z., Yang, J., Zhao, Y., and Liu, J. Sparse-gan: Sparsity-constrained generative adversarial network for anomaly detection in retinal oct image. In *IEEE International Symposium on Biomedical Imaging*, pp. 1227–1231, 2020.