# An extended asymmetric sigmoid with Perceptron(SIGTRON) for imbalanced linear classification 

Hyenkyun Woo


#### Abstract

This article presents a new polynomial parameterized sigmoid called SIGTRON, which is an extended asymmetric sigmoid with Perceptron, and its companion convex model called SIGTRON-imbalanced classification (SIC) model that employs a virtual SIGTRON-induced convex loss function. In contrast to the conventional $\pi$-weighted costsensitive learning model, the SIC model does not have an external $\pi$-weight on the loss function but has internal parameters in the virtual SIGTRON-induced loss function. As a consequence, when the given training dataset is close to the well-balanced condition considering the (scale-)class-imbalance ratio, we show that the proposed SIC model is more adaptive to variations of the dataset, such as the inconsistency of the (scale-)class-imbalance ratio between the training and test datasets. This adaptation is justified by a skewed hyperplane equation, created via linearization of the gradient satisfying $\epsilon$-optimal condition.

Additionally, we present a quasi-Newton optimization(L-BFGS) framework for the virtual convex loss by developing an interval-based bisection line search. Empirically, we have observed that the proposed approach outperforms (or is comparable to) $\pi$-weighted convex focal loss and balanced classifier LIBLINEAR(logistic regression, SVM, and L2SVM) in terms of test classification accuracy with 51 two-class and 67 multi-class datasets. In binary classification problems, where the scale-class-imbalance ratio of the training dataset is not significant but the inconsistency exists, a group of SIC models with the best test accuracy for each dataset (TOP1) outperforms LIBSVM(C-SVC with RBF kernel), a well-known kernel-based classifier.


## Index Terms

Extended exponential function, extended asymmetric sigmoid function, SIGTRON, Perceptron, logistic regression, large margin classification, imbalanced classification, class-imbalance ratio, scale-class-imbalance ratio, line search, Armijo condition, Wolfe condition, quasi-Newton, L-BFGS

## I. InTRODUCTION

Learning a hyperplane from the given training dataset $\mathcal{D}=\left\{\left(x_{l}, y_{l}\right) \in \mathbb{R}^{s} \times\{-1,+1\} \mid l=1,2, \cdots, d\right\}$ is the most fundamental process while we characterize the inherent clustered structure of the test dataset. The main hindrance of the process is that the dataset is imbalanced [14], [27], [42], and there is an inconsistency between the training and test datasets [2]. To address the class-imbalance problem, one can apply under-sampling or oversampling strategies while preserving the cluster structure of dataset $\mathcal{D}$ [21]. In addition to the class imbalance problem, there is another imbalance problem, known as scale imbalance, between the positive class $\left\{x_{i} \mid i \in \mathcal{N}_{+}\right\}$ and the negative class $\left\{x_{j} \mid j \in \mathcal{N}_{-}\right\}$of $\mathcal{D}$ [42]. Here, $\mathcal{N}_{+}=\left\{\ell \mid y_{\ell}=+1\right.$ and $\left.\left(x_{\ell}, y_{\ell}\right) \in \mathcal{D}\right\}$ and $\mathcal{N}_{-}=\left\{\ell \mid y_{\ell}=\right.$ -1 and $\left.\left(x_{\ell}, y_{\ell}\right) \in \mathcal{D}\right\}$. Considering scale and class imbalance simultaneously, we generalize the class-imbalance ratio $r_{c}=\frac{\left|\mathcal{N}_{+}\right|}{\left|\mathcal{N}_{-}\right|}$to the scale-class-imbalance ratio

$$
\begin{equation*}
r_{s c}=r_{c} \sqrt{\frac{\left\|x_{p}^{c}\right\|^{2}+1}{\left\|x_{n}^{c}\right\|^{2}+1}} \tag{1}
\end{equation*}
$$

where $x_{p}^{c}=\frac{1}{\left|\mathcal{N}_{+}\right|} \sum_{i \in \mathcal{N}_{+}} x_{i}$ is the centroid of the positive class of $\mathcal{D}$ and $x_{n}^{c}=\frac{1}{\left|\mathcal{N}_{-}\right|} \sum_{j \in \mathcal{N}_{-}} x_{j}$ is the centroid of the negative class of $\mathcal{D}$. When $r_{s c}=1$ and $\left\|x_{p}^{c}-x_{n}^{c}\right\|>a$ where $a$ is a positive constant, we say that $\mathcal{D}$ is wellbalanced with respect to $r_{s c}$. See [21], [42] for more details on imbalanced problems appearing in classification.

[^0]It is worth mentioning that we can improve the scale imbalance through various normalization methods [16], [23]. In our experiments, we use the well-organized datasets in [8]. They are normalized in each feature dimension with mean zero and variance one. This dimension-wise standardization is the first step of batch normalization [23] in deep learning. By way of the mean zero normalization, the scale-class-imbalanced ratio is always better than the class-imbalance ratio. The details are following.
Theorem I.1. For $\left\{x_{i} \mid i \in \mathcal{N}_{+}\right\}$and $\left\{x_{j} \mid j \in \mathcal{N}_{-}\right\}$, let us consider mean zero normalization, $\sum_{i \in \mathcal{N}_{+}} x_{i}+$ $\sum_{j \in \mathcal{N}_{-}} x_{j}=0$. Here $x_{p}^{c} \neq 0$ and $x_{n}^{c} \neq 0$. Then, if $r_{s c}=1$, we have $r_{c}=1$. On the other hand, if $r_{s c} \neq 1$, we have $\left|r_{s c}-1\right|<\left|r_{c}-1\right|$.
Proof. From $\sum_{i \in \mathcal{N}_{+}} x_{i}=-\sum_{j \in \mathcal{N}_{-}} x_{j}$, we have $r_{c} x_{p}^{c}=-x_{n}^{c}$ and thus $r_{c}=\frac{\left\|x_{n}^{c}\right\|}{\left\|x_{p}^{c}\right\|}$. By using (1) and $r_{c}=\frac{\left\|x_{n}^{c}\right\|}{\left\|x_{p}^{c}\right\|}$, we have $\left\|x_{n}^{c}\right\|^{2}\left(r_{s c}^{2}-1\right)=r_{c}^{2}-r_{s c}^{2}$ and thus the results are obtained.

In addition to the mean zero condition, the variance one condition implies that $r_{c}\left(\frac{\sum_{i \in \mathcal{N}_{+}} x_{i}^{2}}{\left|\mathcal{N}_{+}\right|}-1\right)+\left(\frac{\sum_{j \in \mathcal{N}_{-}} x_{j}^{2}}{\left|\mathcal{N}_{-}\right|}-1\right)=$ 0 . By assuming that $\left|\mathcal{N}_{+}\right|$and $\left|\mathcal{N}_{-}\right|$are sizable, we have $r_{c}\left(\operatorname{Var}\left(\mathcal{N}_{+}\right)+\left(x_{p}^{c}\right)^{2}-1\right)=\left(1-\operatorname{Var}\left(\mathcal{N}_{-}\right)-\left(x_{n}^{c}\right)^{2}\right)$ where $\operatorname{Var}\left(\mathcal{N}_{+}\right)=\frac{1}{\left|\mathcal{N}_{+}\right|} \sum_{i \in \mathcal{N}_{+}}\left(x_{i}-x_{p}^{c}\right)^{2}$ and $\operatorname{Var}\left(\mathcal{N}_{-}\right)=\frac{1}{\left|\mathcal{N}_{-}\right|} \sum_{j \in \mathcal{N}_{-}}\left(x_{j}-x_{n}^{c}\right)^{2}$. If the number of positive instances is sufficiently larger than the number of negative instances, then, from the dimension-wise standardization, we have $x_{p}^{c} \gtrsim 0$ and $\operatorname{Var}\left(\mathcal{N}_{+}\right) \approx 1$. On the contrary, $x_{n}^{c}$ is located away from 0 , and thus $\operatorname{Var}\left(\mathcal{N}_{-}\right)$is relatively small. We have the opposing situation in the case $r_{c} \ll 1$. We notice that when $r_{c}$ is sufficiently far from 1 , the minority class is tiny and far from coordinate zero. It is hard to quantify the variance of the minority class. Thus, the corresponding decision boundary is somehow ambiguous. Even though a deep neural network is trained to the limit with batch normalization, it is hard to obtain reasonable classification results. This phenomenon is known as minority collapse [13]. This article is mainly interested in the case where $r_{s c}$ is not severe, but there exists an inconsistency of $r_{s c}$ between the training and test datasets [2].

In cost-sensitive learning [2], [7], [14], [21], [31], we usually use the $\pi$-weighted loss function to learn a hyperplane decision boundary considering $r_{c}$ of the training dataset. Empirically, $\pi$-weight is set to be in proportion to the inverse of the size of each class. Additionally, it could be determined by the cross-validation [31]. However, it is still unclear how $\pi$-weight is related to $r_{c}$ of the training dataset. [7] uses the inverse of the effective number of instances, known as the expected volume of instances, for the $\pi$-weight. For the dimension-wise standardized dataset [8], the inverse of the proposed $r_{s c}$ (1) is a possible option for $\pi$-weight. For imbalanced object detection, [31] has proposed the famous non-convex focal loss function, which uses the power function of the probability related to the opposite class (and $\pi$-weight) on the logistic loss. They also suggested a $\pi$-weighted convex focal loss function based on standard logistic loss. Regarding test classification accuracy, the convex approach is comparable to the non-convex focal loss. Additionally, see [36], [45] for designing large-margin loss functions and the corresponding $\pi$-weighted cost-sensitive loss functions based on Bregman-divergence.

This article shows the connection between the class-imbalance ratio (including the scale-class-imbalance ratio, the ratio of the effective number of instances, etc) and the loss function via a skewed hyperplane equation. Instead of the margin of each instance, the statistic of the margin distribution [11], [26], i.e., mean-margin, is considered while we describe the skewed hyperplane equation. One of the primary goals of this article is to suggest not a $\pi$-weighted loss function but a new class of adjustable convex loss functions by way of virtualization for novel cost-sensitive learning. For this purpose, we design SIGTRON(extended asymmetric sigmoid with Perceptron) and a novel cost-sensitive learning model, the SIGTRON-imbalanced classification (SIC) model. The proposed SIC model has internal polynomial parameters in the virtual SIGTRON-induced loss function instead of the external $\pi$-weight on the loss function. By the inherent internal structure of the parameters, when $r_{s c}$ (or $r_{c}$ ) of the training dataset is not severe, the SIC model is more adaptable to inconsistencies of (scale-)class-imbalance ratio between training and test datasets. We demonstrate the effectiveness of our model by conducting experiments on 51 two-class datasets. For more information, refer to Figure 6(a) in Section V-A and Table V in Appendix B for $r_{s c}$ and $r_{c}$ of the binary class datasets.

Before we go further, we present the definition of virtualization. The virtual convex loss function $\ell$ is defined as a function satisfying $\nabla \ell=-p$ for the given probability function $p$. For instance, the gradient of the logistic loss function is the negative canonical sigmoid (probability function) $\nabla \ell(x)=-\sigma(-x)$. Various variants of soft-max
function and canonical sigmoid function, such as sparsemax [35], sphericalmax [43], Taylormax [3], high-order sigmoid function [58], and other diverse activation functions [10] are in the category of gradients of virtual loss functions. SIGTRON, which we will introduce in the coming Section IT is also in this category. Although, in this article, we only consider S-shaped probability functions [39], [48] for virtualization, they could be expandable to general functions. A typical example is the quasi-score function, of which the virtual loss function is the negative quasi-likelihood function defined by the mean and variance relation [37], [56], [60]. In addition, virtual loss functions with monotonic gradient function include various ready-made adjustable convex loss functions, such as tunable loss function [30], [50], high-order hinge loss [10], [12], [24], [32], and Logitron [58].
The other main goal of this article is to introduce a quasi-Newton optimization framework for cost-sensitive learning, including the proposed SIC model and $\pi$-weighted convex focal loss [31]. We name the presented optimization framework quasi-Newton(L-BFGS) optimization for virtual convex loss. In quasi-Newton(L-BFGS) optimization, the Hessian matrix is approximated by a rank-two symmetric and positive definite matrix, and its inverse matrix is algorithmically computed by simple two-loop iterations with $m$ recent elements. It generally uses sophisticated cubic-interpolation-based line search to keep positive definiteness. This line search heavily depends on the evaluation of loss function [41], [47]. Instead of the well-known cubic-interpolation-based line search, we propose a relatively simple but accurate line search method, the interval-based bisection line search. With the relatively accurate strong Wolfe stopping criterion, the proposed method performs better than L-BFGS with the cubic-interpolation-based line search regarding test classification accuracy. Please refer to the details in Figure 4 , Although we only consider virtual convex loss functions, which are smooth and bounded below, the proposed optimization framework could be extended to deep neural networks where non-convexity of loss functions is not severe [40]. It is worth mentioning that with the exact line search condition, the nonlinear conjugate gradient utilizes a larger subspace for Hessian matrix approximation [19], [20], [41].
We justify the performance advantage of the proposed approach, the cost-sensitive SIC model and quasi-Newton(L-BFGS) for virtual convex loss, with 118 various classification datasets [8], [58]. For binary classification problems(51 datasets) where $r_{s c}$ of training datasets is not severe, the test classification accuracy of TOP1(a group of SIC models having the best test accuracy for each dataset) is $83.96 \%$, which is $0.74 \%$ better than that of kernelbased LIBSVM(C-SVS with RBF kernel) and $0.16 \%$ better than that of TOP1-FL of $\pi$-weighted convex focal loss. Within linear classifiers, the $\operatorname{MaxA}\left(\alpha_{+}=\frac{7}{8}, \alpha_{-}=\frac{8}{7}\right)$ SIC model shows better performance than the $\pi$-weighted convex focal loss [31] and the balanced classifier LIBLINEAR(logistic regression, SVM, and L2SVM) [12], [15] in terms of test classification accuracy with all 118 datasets. Last but not least, the proposed SIC model with ( $\alpha_{+}, \alpha_{-}$)matrix parameters is a useful tool for understanding the structure of each dataset. For example, see Figure 3 spectf dataset for $r_{s c}$-inconsistency between training and test datasets. The training dataset of spectf is well-balanced, but the test dataset of it is imbalanced [29]. For the multi-label structure, refer to Figure 8 (e) energy-yl dataset and (f) energy-y2 dataset. They have the same input but opposite outputs, such as heating load vs. cooling load [51].

## A. Notation

We briefly review the extended exponential function [59] and the extended logarithmic function [57]. For information on the Tweedie statistical distribution and beta-divergence based on extended elementary functions, refer to the following citations: [1], [28], [57], [59], [60].

For notational convenience, let $\mathbb{R}_{\geq a}=\{x \in \mathbb{R} \mid x \geq a\}$ and $\mathbb{R}_{>a}=\{x \in \mathbb{R} \mid x>a\}$, where $a \in \mathbb{R}$. In the same way, $\mathbb{R}_{\leq a}$ and $\mathbb{R}_{<a}$ are set. Then the extended logarithmic function $\ln _{\alpha, c}$ [57] and the extended exponential function $\exp _{\alpha, c}$ [59] are defined as follows:

$$
\begin{align*}
\ln _{\alpha, c}(x) & = \begin{cases}\ln \left(\frac{x}{c}\right), & \text { if } \alpha=1 \\
c_{\alpha}-x_{\alpha}, & \text { otherwise }\end{cases}  \tag{2}\\
\exp _{\alpha, c}(x) & = \begin{cases}c \exp (x), & \text { if } \alpha=1 \\
c\left(1-\frac{x}{c_{\alpha}}\right)^{1 /(1-\alpha)}, & \text { otherwise }\end{cases} \tag{3}
\end{align*}
$$

where $c>0, \alpha \geq 0, x_{\alpha}=\frac{1}{\alpha-1} x^{1-\alpha}$ and $c_{\alpha}=\frac{1}{\alpha-1} c^{1-\alpha}$. Note that we also explain $c_{\alpha}$ as $(c)_{\alpha}$ for notational convenience. In the case where $c=1$, the extended functions $\exp _{\alpha, c}$ and $\ln _{\alpha, c}$ become the generalized exponential and logarithmic functions [1], [9], [50], respectively. For the effective domains of $\ln _{\alpha, c}$ and $\exp _{\alpha, c}$, see [46], [57], [59]. In this article, we only consider restricted domains of $\ln _{\alpha, c}$ and $\exp _{\alpha, c}$ in Table $\rceil$. Within the restricted domains
in Table I. irrespective of $\alpha_{i}$ and $c_{i}$, we have $\ln _{\alpha_{2}, c_{2}}\left(\exp _{\alpha_{1}, c_{1}}(x)\right) \in \mathbb{R}$ for all $x \in \operatorname{int}\left(\operatorname{dom}\left(\exp _{\alpha_{1}, c_{1}}\right)\right)$. This property defines the extended logistic loss, including high-order sigmoid function [58]. Here, int $(E)$ means the largest open interval contained in an interval $E \subseteq \mathbb{R}$. Note that $\langle x, y\rangle=\sum_{l=1}^{s} x_{l} y_{l}$ for $x, y \in \mathbb{R}^{s},\|x\|=\sqrt{\langle x, x\rangle}$, and $\|x\|_{\infty}=\max _{l}\left|x_{l}\right|$. Additionally, $|\cdot|$ means the absolute value or the size of a discrete set, depending on the context in which it is used.

|  | $\alpha=1$ | $0 \leq \alpha<1$ | $\alpha>1$ |
| :---: | :---: | :---: | :---: |
| $\operatorname{dom}\left(\ln _{\alpha, c}\right)$ | $\mathbb{R}_{>0}$ | $\mathbb{R}_{\geq 0}$ | $\mathbb{R}_{>0}$ |
| $\operatorname{dom}\left(\exp _{\alpha, c}\right)$ | $\mathbb{R}$ | $\mathbb{R}_{\geq c_{\alpha}}$ | $\mathbb{R}_{<c_{\alpha}}$ |
| $\operatorname{dom}\left(\sigma_{\alpha, c}\right)$ | $\mathbb{R}$ | $\mathbb{R}_{\leq-c_{\alpha}}$ | $\mathbb{R}_{\geq-c_{\alpha}}$ |

TABLE I: Restricted domains of the extended logarithmic function $\ln _{\alpha, c}(x)(2)$, the extended exponential function $\exp _{\alpha, c}(x)$ (3), and the extended asymmetric sigmoid function $\sigma_{\alpha, c}(x)$ (6). Here $c>0$. Note that, when $\alpha>1$, $\operatorname{dom}\left(\sigma_{\alpha, c}\right)$ is a closed set $\mathbb{R}_{\geq-c_{\alpha}}$ in the sense $\sigma_{\alpha, c}\left(-c_{\alpha}\right)=\lim _{x \searrow-c_{\alpha}} \frac{c}{c+\exp _{\alpha, c}(-x)}=0 \in \mathbb{R}$.

## B. Cost-sensitive Learning framework and Overview

Let us start with the cost-sensitive learning model

$$
\begin{equation*}
h^{*}=\underset{h \in \mathcal{H}}{\arg \min } \sum_{i \in \mathcal{N}_{+}} L_{+}\left(h\left(x_{i}\right)\right)+\sum_{j \in \mathcal{N}_{-}} L_{-}\left(-h\left(x_{j}\right)\right)+\frac{\lambda}{2} \operatorname{Reg}(w) \tag{4}
\end{equation*}
$$

where $\mathcal{H}=\left\{\langle w, \cdot\rangle+b \mid(w, b) \in \mathbb{R}^{s} \times \mathbb{R}\right\}$ and Reg is an appropriate regularizer for $w$, such as $\|w\|^{2}$. Note that $L_{+}$ and $L_{-}$are virtualized large-margin convex loss functions that are differentiable and lower-bounded. This article only considers the case that $-\nabla L_{ \pm}=p_{ \pm} \in[0,1]$ are probability functions. For more information on cost-sensitive learning, please refer to [2], [14], [16], [21], [27], [31], [42].

In Section II, we study the various properties of SIGTRON, such as smoothness, inflection point, probability-half point, and parameterized mirror symmetry of inflection point with respect to the probability-half point. SIGTRON is used to exemplify the probability functions $p_{ \pm}$in (4). The details of the SIC model are discussed in Section III, where the virtual SIGTRON-induced loss function is introduced. In Section III-A, we derive the skewed hyperplane equation of the cost-sensitive learning model (4), based on the SIGTRON probability function. In Section IV, we demonstrate the usefulness of quasi-Newton optimization(L-BFGS) for virtual convex loss, which includes the interval-based bisection line search. With this optimization method, we solve two different types of cost-sensitive learning models: the SIC model and the $\pi$-weighted convex focal loss. The performance evaluation of the proposed framework, i.e., the SIC model and quasi-Newton optimization(L-BFGS) for virtual convex loss, is done in Section V. We compare the proposed framework with the imbalanced classifier $\pi$-weighted convex focal loss [31], the balanced classifier LIBLINEAR(logistic regression, SVM, and L2SVM) [12], [15], and the nonlinear classifier LIBSVM(C-SVC with RBF kernel) [6]. The conclusion is given in Section VI.

## II. SIGTRON: EXTENDED ASYMMETRIC SIGMOID WITH PERCEPTRON

In this Section, we define SIGTRON using the extended exponential function $\exp _{\alpha, c}$ (3). We then study various properties of SIGTRON, such as its smoothness, inflection point, probability-half point, and parameterized mirror symmetry of the inflection point with respect to the probability-half point.
Definition II. 1 (SIGTRON). Let $\alpha \geq 0, c>0$, and $x \in \mathbb{R}$. Then SIGTRON(extended asymmetric sigmoid with Perceptron) is defined as

$$
s_{\alpha, c}(x)= \begin{cases}\sigma_{\alpha, c}(x) & \text { if } x \in \operatorname{dom}\left(\sigma_{\alpha, c}\right)  \tag{5}\\ \sigma_{P}(x) & \text { otherwise }\end{cases}
$$

where $\sigma_{\alpha, c}$ is the extended asymmetric sigmoid function

$$
\begin{equation*}
\sigma_{\alpha, c}(x)=\frac{c}{c+\exp _{\alpha, c}(-x)} \tag{6}
\end{equation*}
$$

Here, $\exp _{\alpha, c}$ is the extended exponential function (3) and $\sigma_{P}$ is the Perceptron function(or Heaviside function): $\sigma_{P}(x)=1$, if $x \geq 0$ and 0 , otherwise. The restricted domains of $\exp _{\alpha, c}$ and $\sigma_{\alpha, c}$ are defined in Table $I$ Note


Fig. 1: (a) SIGTRON $s_{\alpha, c}(x)$ with $\alpha=\frac{k-1}{k}<1(k=1,2,3,4,6,10)$ and $c_{\alpha}=-1$. Note that $s_{\alpha, c}(x)=1$, if $x \geq-c_{\alpha}$. (b) SIGTRON $s_{\alpha, c}(x)$ with $\alpha=\frac{k+1}{k}>1(k=1,2,3,4,6,10)$ and $c_{\alpha}=1$. Note that $s_{\alpha, c}(x)=0$, if $x \leq-c_{\alpha}$. (c) $\nabla s_{\alpha, c}(x)$ with $\alpha=\frac{k-1}{k}<1(k=2,3,4,6,10)$ and $c_{\alpha}=-1$. Note that $\nabla s_{\alpha, c}(x)=0$, if $x \geq-c_{\alpha}$. The inflection point $x_{i p}$ is getting close to $-c_{\alpha}=1$ as $\alpha \rightarrow 0$. (d) $\nabla s_{\alpha, c}(x)$ with $\alpha=\frac{k+1}{k}>1(k=2,3,4,6,10)$ and $c_{\alpha}=1$. Note that $\nabla s_{\alpha, c}(x)=0$, if $x \leq-c_{\alpha}$. The inflection point $x_{i p}$ is getting close to $-c_{\alpha}=-1$ as $\alpha \rightarrow 2$.
that $s_{\alpha, c}(x) \in[0,1]$ is a non-decreasing continuous function defined on $\mathbb{R}$ with $\lim _{x \rightarrow-\infty} s_{\alpha, c}(x)=0$ and $\lim _{x \rightarrow+\infty} s_{\alpha, c}(x)=1$. Additionally, $s_{\alpha, c}(0)=1 / 2$, irrespective of $\alpha$ and $c_{\alpha}$. Here $x_{p h}=0$ is denoted as the probability-half point. When $\alpha=1, s_{\alpha, c}(x)=\frac{1}{1+\exp (-x)}$ is the canonical sigmoid function, irrespective of $c$.

Note that SIGTRON with $c=1$ becomes the canonical sigmoid function as $|\alpha-1| \rightarrow 0$, since the extended exponential function with $c=1$ is the generalized exponential function. However, SIGTRON with $\left|c_{\alpha}\right|=1$ becomes a smoothed Perceptron as $|\alpha-1| \rightarrow 0$ and $\alpha \neq 1$. Refer to Figure 1 for additional information.

In the following Theorem II.2, we characterize the smoothness of SIGTRON (5) depending on $\alpha$. The proof of Theorem II. 2 is given in Appendix A
Theorem II.2. For $n=1,2,3, \cdots$, when $\alpha \in\left(1-\frac{1}{n}, 1+\frac{1}{n}\right)$, the $n$-th derivative of $s_{\alpha, c}$ is continuous on $\mathbb{R}$ and expressed as

$$
\nabla^{n} s_{\alpha, c}(x)= \begin{cases}\sum_{k=1}^{n} F_{n, k}(x) & \text { if } x \in \operatorname{dom}\left(\sigma_{\alpha, c}\right)  \tag{7}\\ 0 & \text { otherwise }\end{cases}
$$

where

$$
\begin{equation*}
F_{n, k}(x)=A_{n, k}\left(\frac{1}{1-\alpha}\right) \frac{c \exp _{\alpha, c}^{k-n(1-\alpha)}(-x)}{\left(c+\exp _{\alpha, c}(-x)\right)^{k+1}} \tag{8}
\end{equation*}
$$

and

$$
A_{n, k}\left(\frac{1}{1-\alpha}\right)=(-1)^{n+k} k!\sum_{l=0}^{n}\left[\begin{array}{l}
n  \tag{9}\\
l
\end{array}\right]\left\{\begin{array}{l}
l \\
k
\end{array}\right\}(\alpha-1)^{n-l}
$$

Here, $\left[\begin{array}{c}n \\ l\end{array}\right]$ is the Stirling number of the first kind [17| with the recurrence equation $\left[\begin{array}{c}n \\ l\end{array}\right]=(n-1)\left[\begin{array}{c}n-1 \\ l\end{array}\right]+\left[\begin{array}{c}n-1 \\ l-1\end{array}\right]$, where $n, l \geq 1$. $\left\{\begin{array}{l}l \\ k\end{array}\right\}$ is the Stirling number of the second kind with the recurrence equation $\left\{\begin{array}{c}l \\ k\end{array}\right\}=k\left\{\begin{array}{c}l-1 \\ k\end{array}\right\}+$ $\left\{\begin{array}{l}l-1 \\ k-1\end{array}\right\}$, where $l, k \geq 1$.

For the computation of the Stirling number of the first kind and the second kind, we need additional notational conventions: $\left\{\begin{array}{l}0 \\ 0\end{array}\right\}=\left[\begin{array}{l}0 \\ 0\end{array}\right]=1$ and $\left\{\begin{array}{l}a \\ 0\end{array}\right\}=\left[\begin{array}{l}a \\ 0\end{array}\right]=0$ for $a \geq 1$. We have $\left\{\begin{array}{l}a \\ 1\end{array}\right\}=1$ and $\left[\begin{array}{l}a \\ 1\end{array}\right]=(a-1)$ ! with 0 ! $=1$, for $a \geq 1$. Additionally, we note that $\left\{\begin{array}{l}a \\ b\end{array}\right\}=\left[\begin{array}{l}a \\ b\end{array}\right]=0$ if $b>a \geq 0$. For more details, refer to [17].

Theorem II. 2 states that for any $\alpha \in(0,2)$, the gradient of $s_{\alpha, c}(x)$ is given by $c^{\alpha-1}\left(1-s_{\alpha, c}(x)\right)^{\alpha}\left(s_{\alpha, c}(x)\right)^{2-\alpha}$, where $x \in \mathbb{R}$. Check Figure 1 (c) and (d) for a visual representation of $\nabla s_{\alpha, c}(x)$. The information regarding the inflection point of $s_{\alpha, c}$ is provided in Corollary II.3. Additionally, we have observed that the function $\nabla s_{\alpha, c}(x)$ takes the form of the beta distribution $\beta_{D}(x ; \alpha)=\frac{6}{\Gamma(3-\alpha) \Gamma(1+\alpha)} x^{2-\alpha}(1-x)^{\alpha}$, where $x \in[0,1]$. The cumulant distribution of the beta distribution, which has an adjustable parameter $\alpha$, can also be classified as an S -shaped sigmoid function.

Corollary II.3. For $\alpha \in(0,2)$, the inflection point $x_{i p}$ of SIGTRON $s_{\alpha, c}$ exists in the interval int $\left(\operatorname{dom}\left(\sigma_{\alpha, c}\right)\right)$ and is expressed as

$$
x_{i p}=-\ln _{\alpha, c}\left(\frac{c \alpha}{2-\alpha}\right)
$$

When $\alpha=1$, the inflection point is the probability-half point, that is, $x_{i p}=x_{h p}=0$.
Proof. From (7) and Appendix A. we know that $s_{\alpha, c} \in C^{\infty}\left(\operatorname{int}\left(\operatorname{dom}\left(\sigma_{\alpha, c}\right)\right)\right)$ and $\nabla^{2} s_{\alpha, c}(x)=\frac{-\alpha c \exp _{\alpha, c}^{2 \alpha-1}(-x)}{\left(c+\exp _{\alpha, c}(-x)\right)^{2}}+$ $\frac{2 c \exp _{\alpha, c}^{2 \alpha}(-x)}{\left(c+\exp _{\alpha, c}(-x)\right)^{3}}$. Let $\alpha \neq 1$, then, since $\exp _{\alpha, c}(-x) \neq 0$ for $x \in \operatorname{int}\left(\operatorname{dom}\left(\sigma_{\alpha, c}\right)\right)$, the inflection point $x_{i p}$ is a point satisfying $x_{i p}=-\ln _{\alpha, c}\left(\frac{c \alpha}{2-\alpha}\right)$. If $\alpha=1$, then $s_{\alpha, c}$ is the canonical sigmoid function. Thus, $x_{i p}=x_{h p}=0$.

Figure 1 shows $s_{\alpha, c}$ and its derivative for various choices of $\alpha$ satisfying $|\alpha-1|=\frac{1}{k}(k=1,2,3,4,6,10)$ and $\left|c_{\alpha}\right|=1$. Note that $\nabla s_{\alpha, c}$ is not defined at $\alpha=0$ and $\alpha=2$. When $\alpha>1$, the inflection point $x_{i p}$ is getting close to -1 as $\alpha \rightarrow 2$. On the other hand, when $\alpha<1$, the inflection point $x_{i p}$ is getting close to 1 as $\alpha \rightarrow 0$.

Remark II.4. SIGTRON is a general framework for replacing the $S$-shaped sigmoid function in diverse machine learning problems requiring adjustability of probability(or inflection point) and fixed probability-half point. For instance, refer to the skewed hyperplane equation for classification (22) and Example $\overline{I I} .5$ As canonical sigmoid function $\sigma(x)=\frac{1}{1+\exp (-x)}$ has a symmetric property $\sigma(x)=1-\sigma(-x)$, SIGTRON $s_{\alpha, c}$ also has an extended symmetric property:

$$
\begin{equation*}
s_{\alpha, c}(x)=1-s_{2-\alpha, c^{-1}}(-x) \tag{10}
\end{equation*}
$$

where $\alpha \in[0,2]$. Also, for $\alpha \in(0,2)$, we have $\nabla s_{\alpha, c}(x)=\nabla s_{2-\alpha, c^{-1}}(-x)$, the parameterized mirror symmetry with respect to probability-half point $x_{h p}=0$. See Figure 1 (c) and (d) for examples of parameterized mirror symmetry of $\nabla s_{\alpha, c}$. It is worth commenting that the gradient of Logitron $L_{\alpha, c}[58]$ is also a negative probability function, of which the probability-half point depends on $\alpha$. For $\alpha \in(0,2]$, we have

$$
\begin{equation*}
\nabla L_{\alpha, c}(x)=-\left(s_{2-\alpha, c^{-1}}(-x)\right)^{\alpha} \tag{11}
\end{equation*}
$$

where the exponent $\alpha$ is an acceleration parameter of SIGTRON $s_{2-\alpha, c^{-1}}(-x)$ and 10 is used.
Example II.5. It is well-known that it is hard to give a probability for the results of max-margin SVM classifier [33], [44]. In fact, [44] uses the canonical sigmoid function $\sigma(\gamma x+\xi)$ to fit a probability to the classified results of the SVM. Here $\gamma$ and $\xi$ should be estimated [6]. Instead of fitting with the canonical sigmoid function $\sigma(\gamma x+\xi)$, we could use SIGTRON $s_{\alpha, c}$ as a probability estimator for the results of the SVM classifier or any other classifiers having decision boundary, such as hyperplane. For this purpose, there are three steps to follow. First, we must place the probability-half point $x_{h p}$ of $s_{\alpha, c}$ at the decision boundary. Second, we should adjust $c_{\alpha}$ to place the exact probability-one point of $s_{\alpha, c}$ at a specific point, such as the maximum margin point. Finally, we only need to estimate $\alpha$ for the decreasing slope of $s_{\alpha, c}$ based on the distribution of classified results. See [18] for the probability estimation issues in deep neural networks.

## III. Virtual Sigtron-induced loss function, SIC(SIGTRON-Imbalanced Classification) model, AND SKEWED HYPERPLANE EQUATION

This Section studies the SIC model with the virtual SIGTRON-induced loss functions and the skewed hyperplane equation of the SIC model.
Definition III.1. Let $\alpha \in[0,2], c>0$, and $x \in \mathbb{R}$, then the virtual SIGTRON-induced loss function $L_{\alpha, c}^{S}$ is defined by the following gradient equation

$$
\begin{equation*}
\nabla L_{\alpha, c}^{S}(x)=s_{\alpha, c}(x)-1 \tag{12}
\end{equation*}
$$

where $s_{\alpha, c}(x)-1$ is a negative probability function. By the extended symmetric property of SIGTRON in (10), we have $s_{\alpha, c}(x)-1=-s_{2-\alpha, c^{-1}}(-x)$.

We notice that an expansion of the class of Logitron loss (11) via virtualization is easily achieved by $\nabla L_{\beta, \alpha, c}(x)=$ $-\left(s_{2-\alpha, c^{-1}}(-x)\right)^{\beta}$ where $\beta>0$ is a tuning parameter which controls the location of probability-half point $x_{h p}$. Thus, the virtualized Logitron loss contains both the virtual SIGTRON-induced loss (12) and the Logitron loss (11).


Fig. 2: Graphs of the virtual SIGTRON-induced loss function $L_{\alpha, c}^{S}$ for (a) $\alpha=\frac{k-1}{k}$ with $c_{\alpha}=-1$ and (b) $\alpha=\frac{k+1}{k}$ with $c_{\alpha}=1$. Here $k=1,2,4,6$, and 10 . In the case of $k=1,2,4,6, L_{\alpha, c}^{S}$ has a closed-form expression. See Example III. 3

Lemma III.2. Let $\alpha \in[0,1) \cup(1,2]$ and $c>0$. Then the virtual SIGTRON-induced loss function $L_{\alpha, c}^{S}$ satisfying (12) has the following integral formulations:
(1) Case $\alpha \in(1,2]$ :

$$
L_{\alpha, c}^{S}(x)= \begin{cases}-c_{\alpha} F\left(1+\frac{x}{c_{\alpha}} ; \alpha-1\right)+c_{\alpha} & \text { if } x \geq-c_{\alpha}  \tag{13}\\ -x & \text { otherwise } .\end{cases}
$$

(2) Case $\alpha \in[0,1)$ :

$$
L_{\alpha, c}^{S}(x)= \begin{cases}c_{\alpha} F\left(1+\frac{x}{c_{\alpha}} ; 1-\alpha\right)-c_{\alpha}-x & \text { if } x \leq-c_{\alpha}  \tag{14}\\ 0 & \text { otherwise } .\end{cases}
$$

Here, $F(z ; b)=\int_{0}^{z} \frac{1}{1+t^{1 / b}} d t$ with $z \in \mathbb{R}_{\geq 0}$ and $b>0$.
Proof. (1) Case $\alpha \in(1,2]$ : From (12), we have

$$
\nabla L_{\alpha, c}^{S}(x)=\left\{\begin{array}{lr}
-\frac{1}{1+\left(1+\frac{x}{c_{\alpha}}\right)^{\frac{1}{\alpha-1}}} & \text { if } x \geq-c_{\alpha} \\
-1 & \text { otherwise },
\end{array}\right.
$$

where $-c_{\alpha}<0$ and $1+\frac{x}{c_{\alpha}} \geq 0$. The integration of $\nabla L_{\alpha, c}^{S}$ becomes

$$
L_{\alpha, c}^{S}\left(a_{1}\right)-L_{\alpha, c}^{S}\left(a_{0}\right)=\int_{a_{0}}^{a_{1}} \nabla L_{\alpha, c}^{S}(t) d t= \begin{cases}-c_{\alpha} F\left(1+a_{1} / c_{\alpha} ; \alpha-1\right)+c_{\alpha}+a_{0}, & \text { if } a_{1} \geq-c_{\alpha} \\ -a_{1}+a_{0}, & \text { otherwise },\end{cases}
$$

where we may choose $a_{0} \ll-c_{\alpha}$. Then, we get the virtual SIGTRON-induced loss function (13), after setting $a_{1}=x$ and removing constants.
(2) Case $\alpha \in[0,1)$ : We have

$$
\nabla L_{\alpha, c}^{S}(x)= \begin{cases}\frac{1}{1+\left(1+\frac{x}{c_{\alpha}}\right)^{\frac{1}{1-\alpha}}}-1 & \text { if } x \leq-c_{\alpha} \\ 0 & \text { otherwise },\end{cases}
$$

where $-c_{\alpha}>0$ and $1+\frac{x}{c_{\alpha}} \geq 0$. Thus, we get

$$
L_{\alpha, c}^{S}\left(a_{0}\right)-L_{\alpha, c}^{S}\left(a_{1}\right)=\int_{a_{1}}^{a_{0}} \nabla L_{\alpha, c}^{S}(t) d t= \begin{cases}-c_{\alpha} F\left(1+\frac{a_{1}}{c_{\alpha}} ; 1-\alpha\right)+c_{\alpha}+a_{1} & \text { if } a_{1} \leq-c_{\alpha} \\ 0 & \text { otherwise },\end{cases}
$$

where $a_{0}>-c_{\alpha}$. Let $a_{1}=x$, then we get the virtual SIGTRON-induced loss function (14).
In Figure 2, we present the virtual SIGTRON-induced loss $L_{\alpha, c}^{S}(x)$ with $\left|c_{\alpha}\right|=1$ and $|\alpha-1|=\frac{1}{k}$. Here $k=1,2,4,6$ and 10 are the polynomial orders of $\exp _{\alpha, c}$. For $k=10, L_{\alpha, c}^{S}(x)$ is computed directly by (13) and (14). For $k=1,2,4,6, L_{\alpha, c}^{S}(x)$ is expressed in a closed form by virtue of Example III.3. As we increase the
polynomial order $k=\frac{1}{|\alpha-1|}$, i.e. $\alpha \rightarrow 1$ and $\left|c_{\alpha}\right|=1, L_{\alpha, c}^{S}(x)$ is getting close to the smoothed Perceptron loss function [54], not to the logistic loss.

Example III.3. We make a list of $F(a ; 1 / k)$ for $k=1, \cdots, 6$.

- $k=1: F(a ; 1 / 1)=\ln (1+a)$
- $k=2: F(a ; 1 / 2)=\arctan (a)$
- $k=3: F(a ; 1 / 3)=\frac{1}{6} \log \left(1+\frac{3 a}{a^{2}-a+1}\right)+\frac{1}{\sqrt{3}} \arctan \left(\frac{2 a-1}{\sqrt{3}}\right)-\frac{1}{\sqrt{3}} \arctan \left(-\frac{1}{\sqrt{3}}\right)$
- $k=4: F(a ; 1 / 4)=\frac{1}{4 \sqrt{2}} \log \left(1+\frac{2 \sqrt{2} a}{a^{2}-\sqrt{2} a+1}\right)+\frac{1}{2 \sqrt{2}} \arctan (1+\sqrt{2} a)-2 \arctan (1-\sqrt{2} a)$
- $k=5: F(a ; 1 / 5)=\frac{(\sqrt{5}-1)}{20} \log \left(2 a^{2}+(\sqrt{5}-1) a+2\right)-\frac{(\sqrt{5}+1)}{20} \log \left(2 a^{2}-(\sqrt{5}+1) a+2\right)+\frac{\log (1+a)}{5}$
$-\frac{\sqrt{10-2 \sqrt{5}}}{10} \arctan \left(\frac{-4 a+\sqrt{5}+1}{\sqrt{10-2 \sqrt{5}}}\right)+\frac{\sqrt{10+2 \sqrt{5}}}{10} \arctan \left(\frac{4 a+\sqrt{5}-1}{\sqrt{10+2 \sqrt{5}}}\right)-\frac{(\sqrt{5}-1) \log 2+(\sqrt{5}+1) \log 2}{20}$
$+\frac{\sqrt{10-2 \sqrt{5}}}{10} \arctan \left(\frac{\sqrt{5}+1}{\sqrt{10-2 \sqrt{5}}}\right)-\frac{\sqrt{10+2 \sqrt{5}}}{10} \arctan \left(\frac{\sqrt{5}-1}{\sqrt{10+2 \sqrt{5}}}\right)$
- $k=6: F(a ; 1 / 6)=\sqrt{3} \log \left(\frac{a^{2}+\sqrt{3} a+1}{a^{2}-\sqrt{3} a+1}\right)+\arctan (\sqrt{3}+2 a) / 6-\arctan (\sqrt{3}-2 a) / 6+\arctan (a) / 3$


## A. Learning a hyperplane with SIC model

Let us first consider the cost-sensitive convex minimization model (4) to find a hyperplane $h^{*}(x)=0$ from the given training dataset $\mathcal{D}$. The following is the realization of (4) through the virtual SIGTRON-induced loss function (12) and $\ell_{2}$-regularizer.

$$
\begin{equation*}
h^{*}=\underset{h \in \mathcal{H}}{\arg \min } \mathcal{F}(h) \tag{15}
\end{equation*}
$$

where $\mathcal{H}=\left\{\langle w, \cdot\rangle+b \mid(w, b) \in \mathbb{R}^{s} \times \mathbb{R}\right\}$ and

$$
\begin{equation*}
\mathcal{F}(h)=\sum_{i \in \mathcal{N}_{+}} L_{\alpha_{+}, c_{+}}^{S}\left(h\left(x_{i}\right)\right)+\sum_{j \in \mathcal{N}_{-}} L_{\alpha_{-}, c_{-}}^{S}\left(-h\left(x_{j}\right)\right)+\frac{\lambda}{2}\|w\|_{2}^{2} . \tag{16}
\end{equation*}
$$

This minimization problem (15) with (16) is named as the SIGTRON-imbalanced classification(SIC) model. In the following example, we introduce a feature of the SIC model which is similar to the SVM.

Example III.4. Let us assume that the training dataset $\mathcal{D}$ is separable, $\alpha_{+}, \alpha_{-} \in[0,1), \lambda=0$, and $\left\|w^{*}\right\|=1$. When $\mathcal{F}\left(h^{*}\right)=0$, from Lemma III.2 we have $h^{*}\left(x_{i}\right) \geq-\left(c_{+}\right)_{\alpha_{+}}$for all $x_{i} \in \mathcal{N}_{+}$and $h^{*}\left(x_{j}\right) \leq\left(c_{-}\right)_{\alpha_{-}}$for all $x_{j} \in \mathcal{N}_{-}$. Let $x_{p}^{m}$ be a point satisfying $h^{*}\left(x_{p}^{m}\right)=\min _{i \in \mathcal{N}_{+}} h^{*}\left(x_{i}\right)$ and $x_{n}^{m}$ be a point satisfying $h^{*}\left(x_{n}^{m}\right)=$ $\max _{j \in \mathcal{N}_{-}} h^{*}\left(x_{j}\right)$. Then, if we set $-\left(c_{+}\right)_{\alpha_{+}}=h^{*}\left(x_{p}^{m}\right)$ and $\left(c_{-}\right)_{\alpha_{-}}=h^{*}\left(x_{n}^{m}\right)$, we have the following skewed hyperplane equation

$$
\begin{equation*}
\left\{x \in \mathbb{R}^{s} \left\lvert\,\left\langle w^{*}, x-\frac{x_{p}^{m}+x_{n}^{m}}{2}\right\rangle=\frac{\left(c_{+}\right)_{\alpha_{+}}-\left(c_{-}\right)_{\alpha_{-}}}{2}\right.\right\} . \tag{17}
\end{equation*}
$$

Here, when $\left(c_{+}\right)_{\alpha_{+}}=\left(c_{-}\right)_{\alpha_{-}}$, we have the max-margin hyperplane $\left\{x \in \mathbb{R}^{s} \left\lvert\,\left\langle w^{*}, x-\frac{x_{\infty}^{m}+x_{n}^{m}}{2}\right\rangle=0\right.\right\}$.
In the max-margin region where $\mathcal{F}\left(h^{*}\right)=0$ with $\lambda=0, r_{c}$ does not affect the location of the hyperplane. We also notice that $\pi$-weight on the loss function is meaningless in this region. As observed in Example III.4, margin parameters $\left(c_{+}\right)_{\alpha_{+}}$and $\left(c_{-}\right)_{\alpha_{-}}$of the SIC model (15) are mainly relevant to the location of the (skewed) hyperplane. See [4], [25], [49] for related issues in deep learning where the max-margin region exists at infinity.

From now on, we are interested in $h^{*}$, satisfying the following $\epsilon$-optimal condition

$$
\begin{equation*}
\left\|\nabla \mathcal{F}\left(h^{*}\right)\right\|_{\infty} \leq \epsilon, \tag{18}
\end{equation*}
$$

where $\mathcal{F}$ is defined in (16). This $\epsilon$-optimal condition (18) is used as a gradient-based stopping criterion while we do numerical experiments in Section V . The gradient of $\mathcal{F}\left(h^{*}\right)$ becomes

$$
\nabla \mathcal{F}\left(h^{*}\right)=\sum_{i \in \mathcal{N}_{+}}-p_{+}\left(h^{*}\left(x_{i}\right)\right)\left[\begin{array}{c}
x_{i}  \tag{19}\\
1
\end{array}\right]+\sum_{j \in \mathcal{N}_{-}} p_{-}\left(-h^{*}\left(x_{j}\right)\right)\left[\begin{array}{c}
x_{j} \\
1
\end{array}\right]+\lambda w^{*} \in[-\epsilon \mathbf{1},+\epsilon \mathbf{1}]
$$

where $p_{+}=-\nabla L_{\alpha_{+}, c_{+}}^{S}, p_{-}=-\nabla L_{\alpha_{-}, c_{-}}^{S} \in(0,1)$, and 1 is all one vector in $\mathbb{R}^{s+1}$. Let us assume that $p_{ \pm}$is twice differentiable and $\nabla p_{ \pm}(0) \neq 0$. Then by linearization of $p_{ \pm}$at $0[11]$, we have $p_{ \pm}(a)=p_{ \pm}(0)+a \nabla p_{ \pm}(0)+D_{p_{ \pm}}(a \mid 0)$ where $D_{p_{ \pm}}(a \mid 0)=\int_{0}^{a} \nabla^{2} p_{ \pm}(y)(a-y) d y$ and thus, in terms of $\left.b^{*}, 19\right)$ becomes
$\sum_{i \in \mathcal{N}_{+}}\left(p_{+}(0)+h^{*}\left(x_{i}\right) \nabla p_{+}(0)+D_{p_{+}}\left(h^{*}\left(x_{i}\right) \mid 0\right)\right)-\sum_{j \in \mathcal{N}_{-}}\left(p_{-}(0)-h^{*}\left(x_{j}\right) \nabla p_{-}(0)+D_{p_{-}}\left(-h^{*}\left(x_{j}\right) \mid 0\right)\right) \in[-\epsilon,+\epsilon]$.
This equation simplifies to

$$
\begin{equation*}
r_{c}\left(p_{+}(0)+h^{*}\left(x_{p}^{c}\right) \nabla p_{+}(0)+E_{+}\right)-\left(p_{-}(0)-h^{*}\left(x_{n}^{c}\right) \nabla p_{-}(0)+E_{-}\right) \in I_{\epsilon} \tag{20}
\end{equation*}
$$

where $E_{+}=\frac{\sum_{i \in \mathcal{N}_{+}} D_{p_{+}}\left(h^{*}\left(x_{i}\right) \mid 0\right)}{\left|\mathcal{N}_{+}\right|}, E_{-}=\frac{\sum_{j \in \mathcal{N}_{-}} D_{p_{-}}\left(-h^{*}\left(x_{j}\right) \mid 0\right)}{\left|\mathcal{N}_{-}\right|}$, and $I_{\epsilon}=\left[\frac{-\epsilon}{\mid \mathcal{N}_{--}}, \frac{+\epsilon}{\left|\mathcal{N}_{-}\right|}\right]$. Since $h^{*}(x)=\left\langle w^{*}, x\right\rangle+b^{*}$, we have an expression below explaining $b^{*}$ from the $\epsilon$-optimal condition (18)

$$
\begin{equation*}
\left\langle w^{*}, A\right\rangle+b^{*} \in-E \tag{21}
\end{equation*}
$$

where $A=\frac{r_{c} \nabla p_{+}(0) x_{p}^{c}+\nabla p_{-}(0) x_{n}^{c}}{r_{c} \nabla p_{+}(0)+\nabla p_{-}(0)}$ and $E=\frac{\left(r_{c} p_{+}(0)-p_{-}(0)\right)+\left(r_{c} E_{+}-E_{-}\right)+I_{\epsilon}}{r_{c} \nabla p_{+}(0)+\nabla p_{-}(0)}$. Now, the skewed hyperplane equation for $h^{*}(x)=0$ is expressed as

$$
\begin{equation*}
\left\{x \in \mathbb{R}^{s} \mid\left\langle w^{*}, x-A\right\rangle \in E\right\} \tag{22}
\end{equation*}
$$

Through this skewed hyperplane equation, we could somehow understand a mysterious connection between the hyperplane decision boundary and the loss function. Depending on the skewness level $E$, the hyperplane decision boundary $h^{*}(x)=0$ may not be located between $x_{p}^{c}$ and $x_{n}^{c}$. The effect of $E$, however, can be discarded in a particular dataset structure. The details are following.

Example III.5. Let us assume that the given dataset $\mathcal{D}$ has a symmetric structure with $r_{c}=1$. That is, there is a one-to-one correspondence between the positive dataset $\mathcal{N}_{+}$and the negative dataset $\mathcal{N}_{-}$with respect to the hyperplane $h^{*}(x)=0$. For any $x_{i} \in \mathbb{R}^{s}$ where $i \in \mathcal{N}_{+}$, there is an unique $x_{j} \in \mathbb{R}^{s}$ where $j \in \mathcal{N}_{-}$, such that $h^{*}\left(x_{i}\right) \approx-h^{*}\left(x_{j}\right)$. The opposite is also true. If we additionally assume that $p_{-}(x)=p_{+}(x)$ then we obtain $E_{+} \approx E_{-}$. Thus, the skewed hyperplane equation (22) becomes

$$
\begin{equation*}
\left\{x \in \mathbb{R}^{s} \mid\left\langle w^{*}, x-A\right\rangle \approx 0\right\} \tag{23}
\end{equation*}
$$

where $A=\frac{x_{r}^{c}+x_{n}^{c}}{2}$.
By using a physically symmetric and separable dataset, [34] analyzes the structure of the Leaky ReLU two-layer neural network in the max-margin region at infinity. Please refer to [53] for a review of the implicit bias, including max-margin region at infinity, in machine learning including deep neural networks. In the following Theorem, we summarize the skewed hyperplane equation obtained by the SIC model (15) under the condition $\left|r_{c} E_{+}-E_{-}\right| \ll 1$.
Theorem III.6. Let $\left\|x_{p}^{c}-x_{n}^{c}\right\|>a$ where $a>0,0<h^{*}\left(x_{p}^{c}\right)<1$, and $0<-h^{*}\left(x_{n}^{c}\right)<1, \alpha_{ \pm} \in[0,1) \cup(1,2]$. Additionally, assume that $\left|r_{c} E_{+}-E_{-}\right| \ll 1$. Then the skewed hyperplane equation (22) becomes

$$
\begin{equation*}
\left\{x \in \mathbb{R}^{s} \left\lvert\,\left\langle w^{*}, x-\left(\frac{r_{c} c_{+}^{\alpha_{+}-1} x_{p}^{c}+c_{-}^{\alpha_{-}-1} x_{n}^{c}}{r_{c} c_{+}^{\alpha_{+}-1}+c_{-}^{\alpha_{-}-1}}\right)\right\rangle \approx \frac{2\left(r_{c}-1\right)}{r_{c} c_{+}^{\alpha_{+}-1}+c_{-}^{\alpha_{-}-1}}\right.\right\} \tag{24}
\end{equation*}
$$

If $r_{c}=1$ then 21) is expressed as $\left\langle w^{*},(1-\eta) x_{p}^{c}+\eta x_{n}^{c}\right\rangle+b^{*} \approx 0$ and the signed distance of $x_{p}^{c}$ to the hyperplane $h^{*}(x)=0$ is approximately given as

$$
\begin{equation*}
\frac{h^{*}\left(x_{p}^{c}\right)}{\left\|w^{*}\right\|} \approx \eta\left\|x_{p}^{c}-x_{n}^{c}\right\| \cos \left(\theta_{+}\right) \tag{25}
\end{equation*}
$$

where $\eta=\frac{c_{-}^{\alpha_{-}-1}}{c_{+}^{\alpha_{+}^{-1}+c_{-}^{\alpha_{-}-1}}} \in(0,1)$ and $\cos \left(\theta_{+}\right)=\left\langle\frac{w^{*}}{\left\|w^{*}\right\|}, \frac{x_{p}^{c}-x_{n}^{c}}{\left\|x_{p}^{c}-x_{n}^{c}\right\|}\right\rangle>0$. In the same way, for $x_{n}^{c}$, we have $\frac{h^{*}\left(x_{n}^{c}\right)}{\left\|w^{*}\right\|} \approx(\eta-1)\left\|x_{p}^{c}-x_{n}^{c}\right\| \cos \left(\theta_{+}\right)$.


Fig. 3: Classification results with the spectf dataset in Table V. The test dataset has $r_{s c}=0.26\left(r_{c}=0.09\right)$. However, the training dataset is well-balanced, i.e., $r_{s c}=1$. We have $20 \times 20$ hyperplanes $h_{\left(\alpha_{+}, \alpha_{-}\right)}^{*}(x)=0$ by solving $20 \times 20$ SIC models (15) with the well-balanced training dataset. (a) The pattern of the test classification accuracy. (b) The pattern of the signed distance of the centroid of the positive test dataset to the hyperplane. (c) The pattern of the signed distance of the centroid of the negative test dataset to the hyperplane. (d) The pattern of $\eta$ in 26. The best test accuracy is achieved at $\left(\alpha_{+}, \alpha_{-}\right)=\left(\frac{11}{10}, 2\right)$. This point is the smallest distance of the centroid of the positive test dataset to the hyperplane. And, it is contained in the group $\left\{\left(\frac{11}{10}, 2\right),\left(\frac{9}{10}, 2\right),\left(\frac{11}{10}, 0\right),\left(\frac{9}{10}, 0\right)\right\}$ having the smallest $\eta=\frac{1}{11}$. See Example III. 8 for more details.

In practice, due to computational constraints, we normally choose polynomial functions for $\exp _{\alpha, c}$, i.e., positive integers for $\frac{1}{\left|\alpha_{ \pm}-1\right|}$. Assume that $\left|\left(c_{+}\right)_{\alpha_{+}}\right|=\left|\left(c_{-}\right)_{\alpha_{-}}\right|$is a constant, $\frac{1}{\left|\alpha_{+}-1\right|}=k_{+}$, and $\frac{1}{\left|\alpha_{-}-1\right|}=k_{-}$. Here, $k_{ \pm}=1,2,3, \cdots$. Then we have

$$
\begin{equation*}
\eta=\frac{1 / c_{-}^{1-\alpha_{-}}}{1 / c_{+}^{1-\alpha_{+}}+1 / c_{-}^{1-\alpha_{-}}}=\frac{k_{-}}{k_{+}+k_{-}} \tag{26}
\end{equation*}
$$

The skewed hyperplane equation (24) is interpreted as that the hyperplane $h^{*}(x)=0$ is tuned by the ratio of polynomial order of SIGTRON if $\cos \left(\theta_{+}\right)$does not change much. We notice that [5] has added a margin-related parameter, of which the role is similar to $c_{\alpha}$ of our SIC model, inside of the loss function to adjust decision boundary in imbalanced deep neural network. Before we go further, we introduce a simplified model to show the role of $r_{s c}$ in the skewed hyperplane equation (24).
Remark III.7. Let us assume that $x_{i} \approx x_{p}^{c}>0$ for $i \in \mathcal{N}_{+}, x_{j} \approx x_{n}^{c}>0$ for $j \in \mathcal{N}_{-}, \lambda=0$, and $p_{ \pm} \in(0,1)$. Then (19) becomes $r_{c} p_{+}\left(h^{*}\left(x_{p}^{c}\right)\right)\left[\begin{array}{c}x_{p}^{c} \\ 1\end{array}\right] \approx p_{-}\left(-h^{*}\left(x_{n}^{c}\right)\right)\left[\begin{array}{c}x_{n}^{c} \\ 1\end{array}\right]$. We apply $\left\langle\cdot,\left[\begin{array}{c}x_{p}^{c} \\ 1\end{array}\right]\right\rangle$ and $\left\langle\cdot,\left[\begin{array}{c}x_{n}^{c} \\ 1\end{array}\right]\right\rangle$. Then we simplify the corresponding equations. As a result, we have $r_{s c} p_{+}\left(h^{*}\left(x_{p}^{c}\right)\right) \approx p_{-}\left(-h^{*}\left(x_{n}^{c}\right)\right)$. After linearization at 0 [11], we obtain (24] with $r_{s c}$, instead of data-sensitive $r_{c}$. Actually, as observed in Theorem I.1] by the dimensionwise standardization of the dataset, we always have $\left|r_{s c}-1\right|<\left|r_{c}-1\right|$ where $r_{s c} \neq 1$, irrespective of the domain of the dataset $\mathcal{D}$. Hence, empirically we could use $r_{s c}$, instead of $r_{c}$ (see Example III.8). In addition, refer to [7] where they invent effective number to replace the role of $r_{c}$ in imbalanced classification.

The following Example III.8 describes the tunable hyperplane via skewed hyperplane equation for $r_{s c}$-inconsistent dataset having the well-balanced training and imbalanced test datasets.
Example III.8. Let us start with the two-class 'spectf' dataset in Table $\square$ The training dataset is well-balanced, i.e., $r_{s c}=1$. However, the test dataset has $r_{s c}=0.26\left(r_{c}=0.09\right)$. It indicates that the positive class of the test dataset is the minority class. The hyperplane to be learned should be located near the minority class to achieve better test classification accuracy. As observed in (25) and Figure 3 (d), to move the hyperplane to the minority class as close as we can, we need to select the smallest $\eta=1 / 11$. This $\eta$ corresponds to four ( $\alpha_{+}, \alpha_{-}$) candidates: $(11 / 10,2),(9 / 10,2),(11 / 10,0)$, and $(9 / 10,0)$. In fact, at $\left(\alpha_{+}, \alpha_{-}\right)=(11 / 10,2)$, we obtain the minimum distance of $x_{\text {test }, p}^{c}$ (the centroid of the positive class of test dataset) to the hyperplane $h_{\left(\alpha_{+}=11 / 10, \alpha_{-}=2\right)}^{*}(x)=0$ (Figure 3 (b)) and the best test classification accuracy $64.6 \%$ (Figure 3 (a)). Note that the pattern of $\eta$ in Figure 3 (d) is similar to the pattern of the distance of $x_{\text {test }, p}^{c}$ to the hyperplane in Figure 3 (b). As Figure 3 (a) shows, the region $\alpha_{-} \approx 2$ obtains better test classification accuracy than the region $\alpha_{-} \approx 0$. Additionally, note that $\cos \left(\theta_{\text {test },+}\right)=\left\langle\begin{array}{c}w_{\left(+, \alpha_{-}\right)}^{*} \\ \left\|w_{\left(\alpha_{+}, \alpha_{-}\right)}^{*}\right\|\end{array}\left\|\frac{x_{\text {test }}^{c}-x_{\text {test }, n}^{c}}{\left\|x_{\text {test }, p}-x_{\text {test }, n}\right\|}\right\|\right\rangle[0.52,0.90]$ and $\mathbf{E}\left(\cos \left(\theta_{\text {test },+}\right)\right)=0.71$. As a reference, we obtained $20 \times 20$ hyperplanes $h_{\left(\alpha_{+}, \alpha_{-}\right)}^{*}(x)=0$ by solving $20 \times 20$ SIC models (15) with the well-balanced training dataset. The cross-validation was used for the best regularization parameter $\lambda$. We set $\left|\left(c_{+}\right)_{\alpha_{+}}\right|=\left|\left(c_{-}\right)_{\alpha_{-}}\right|=2$, $\frac{1}{\left|1-\alpha_{+}\right|}=k_{+}=1,2, \cdots, 10$, and $\frac{1}{\left|1-\alpha_{-}\right|}=k_{-}=1,2, \cdots, 10$.
In Example III.8, we saw that the hyperplane should be located near minority class for better performance. Depending on the pattern of the distribution of the data in the minority class, e.g. the variance of the minority class, it may not be true. For instance, [5] recommend that the decision boundary be required to be placed near majority class. For better understanding of the pattern of the data distribution, we recommend to use ( $\alpha_{+}, \alpha_{-}$)matrix.

Remark III.9. Lately, [37] has proposed two focal loss functions for imbalanced object detection. The first one is the non-convex focal loss function. It has $L_{+}(h(x))=-\pi(1-p(h(x)))^{\gamma_{g}} \log (p(h(x)))$ and $L_{-}(h(x))=-(1-$ $\pi) p(h(x))^{\gamma_{s}} \log (1-p(h(x)))$ where $p(h(x)) \in(0,1)$ is a probability function, like canonical sigmoid $\sigma$ or reduced Sigtron. The second one is the convex focal loss function. It has $L_{+}(h(x))=-\pi \log (\sigma(\gamma h(x)+\xi))$ and $L_{-}(h(x))=$ $-(1-\pi) \log (1-\sigma(\gamma h(x)+\xi))$. Here $\pi \in(0,1)$ is known as a cost-sensitive parameter to be selected depending on $r_{c}\left(o r r_{s} c\right.$ ) of the training dataset. Note that $\gamma \geq 1$ and $\xi \geq 0$ control the stiffness and shift of the convex focal loss, respectively. As [31] mentioned, the performance gap between the two types of focal losses is negligible. Therefore, we exclusively compare the convex focal loss to the convex SIC model. To find additional information, please refer to Section V-A

## IV. Quasi-Newton optimization(L-BFGS) for virtual convex loss

This Section presents quasi-Newton optimization(L-BFGS) for virtual convex loss framework. It includes the proposed interval-based bisection line search, which uses gradients of a virtual convex loss function.

Let us discuss the SIC model (15), where $\mathcal{F}(h)$ is convex, differentiable, and bounded below. It is worth noting that the optimization framework we will be proposing for this model can also be used for cost-sensitive learning model (4), including the $\pi$-weighted convex focal loss. Before we proceed, let us take a moment to review the quasi-Newton optimization framework described in [41]. The iterates $h_{0}, h_{1}, h_{2}, \cdots$ satisfy $h_{t+1}=h_{t}+\rho_{t} z_{t}$ where $\rho_{t}>0$ is a step length and $z_{t}=-B_{t}^{-1} \nabla \mathcal{F}\left(h_{t}\right)$ is a descent direction. Here, $B_{t}$ is a symmetric and positive definite rank-two approximation of the Hessian matrix $\nabla^{2} \mathcal{F}\left(h_{t}\right)$. Interestingly, L-BFGS directly approximates $B_{t}^{-1} \nabla \mathcal{F}\left(h_{t}\right)$ by two-loop iterations with $m$ recent elements. Here, $m$ is the tuning parameter of L-BFGS. The performance comparison of the proposed optimization framework considering $m$ of L-BFGS is shown in Figure 4 For the initial point, we set $h_{0}=0$, corresponding to the probability-half point of SIGTRON in the gradient of the SIC model. It is well known that, to guarantee sufficient descent of $\mathcal{F}(h)$ and positive definiteness of low-rank matrix $B_{t}$, the step length $\rho_{t}$ of L-BFGS should satisfy the Armijo condition (27) and the Wolfe condition (28):

$$
\begin{equation*}
\mathcal{F}\left(h_{t}+\rho_{t} z_{t}\right)-\mathcal{F}\left(h_{t}\right) \leq c_{I} \rho_{t}\left\langle\nabla \mathcal{F}\left(h_{t}\right), z_{t}\right\rangle \tag{27}
\end{equation*}
$$

and

$$
\begin{equation*}
\left\langle\nabla \mathcal{F}\left(h_{t}+\rho_{t} z_{t}\right), z_{t}\right\rangle \geq c_{I I}\left\langle\nabla \mathcal{F}\left(h_{t}\right), z_{t}\right\rangle \tag{28}
\end{equation*}
$$



Fig. 4: A comparison of performanve between quasi-Newton(L-BFGS) for virtual convex loss, which uses the strong Wolfe condition (32) with $c_{I I} \in[0.1,0.9]$, and the classic L-BFGS $\left(^{*}\right)$, which uses the strong Wolfe condition (32) with $c_{I I}=0.9$ and the Armijo condition 27 with $c_{I}=10^{-4}$. Note that L-BFGS $(*)$ uses the cubic-interpolationbased line search [41], [47]. For our experiments, we use $12 \times 12$ SIC models with $\left|c_{\alpha}\right|=1$ and $\left|\alpha_{ \pm}-1\right|=\frac{1}{k}$, where $k=1,2,3,4,5,6$. (a) Mean test accuracy of $12 \times 12$ SIC models. (b) Maximum test accuracy and (c) minimum test accuracy obtained by an SIC model with fixed $\alpha_{ \pm}$for each $c_{I I}$. (d) Test accuracy of TOP1 for each $c_{I I}$. (e) Total computation time of $12 \times 12$ SIC models. Here, we report the average values of five times repeated experiments with all datasets in Table $V$ and VI For $0.1 \leq c_{I I} \leq 0.5$, in terms of mean test classification accuracy in (a), quasi-Newton(L-BFGS) for virtual convex loss outperforms L-BFGS(*), for each $m$ of two loop iterations.
where $0<c_{I}<c_{I I}<1$. The Armijo condition (27) can be reformulated through the expectation of gradients:

$$
\begin{equation*}
\phi\left(\rho_{t}\right)-\phi(0)=\int_{0}^{\rho_{t}} \phi^{\prime}(\rho) d \rho=\rho_{t} \mathbf{E}_{\left[0, \rho_{t}\right]}\left(\phi^{\prime}\right) \tag{29}
\end{equation*}
$$

where $\phi\left(\rho_{t}\right)=\mathcal{F}\left(h_{t}+\rho_{t} z_{t}\right)$ and $\phi^{\prime}(\rho)=\left\langle\nabla \mathcal{F}\left(h_{t}+\rho z_{t}\right), z_{t}\right\rangle$. Note that $\phi^{\prime}(0)=\left\langle\nabla \mathcal{F}\left(h_{t}\right), z_{t}\right\rangle<0$ where $z_{t}=-B_{t}^{-1} \nabla \mathcal{F}\left(h_{t}\right)$. Now, we get the reformulated Armijo condition

$$
\begin{equation*}
\mathbf{E}_{\left[0, \rho_{t}\right]}\left(\phi^{\prime}\right) \leq c_{I} \phi^{\prime}(0) \tag{30}
\end{equation*}
$$

and the Wolfe condition

$$
\begin{equation*}
c_{I I} \phi^{\prime}(0) \leq \phi^{\prime}\left(\rho_{t}\right) \tag{31}
\end{equation*}
$$

where (31) is also known as the curvature condition [38], which is clearly understood by the reformulation of (31) as $\mathbf{E}_{\left[0, \rho_{t}\right]}\left(\phi^{\prime \prime}\right)>\left(c_{I I}-1\right) \rho_{t} \phi^{\prime}(0)>0$. The positive definiteness of $B_{t}$ in L-BFGS is adjusted by $c_{I I} \in(0,1)$, normally set as 0.9 . For more details, see [41].

The reformulated Armijo condition (30) has several advantages, compared to the Armijo condition 27). First, it is more intuitive about the descent condition of the loss function. The average slopes of $\phi$ in the interval $\left[0, \rho_{t}\right]$ must be less than the initial slope $\phi^{\prime}(0)$. Second, for the SIC model (15), using an approximation of (30) is more practical. That is, $\mathbf{E}_{\left[0, \rho_{t}\right]}\left(\phi^{\prime}\right) \approx \sum_{i=1}^{n} a_{i} \phi^{\prime}\left(\tilde{\rho}_{i}\right)$, where $a_{i} \geq 0, \sum_{i=1}^{n} a_{i}=1$, and $0 \leq \tilde{\rho}_{0}<\tilde{\rho}_{1}<\cdots<\tilde{\rho}_{n} \leq \rho_{t}$.

This approach is workable for the general loss function, including virtual non-convex loss function. For a virtual convex loss function, however, we do not need to evaluate a relatively large number of directional derivatives in the interval $\left[0, \rho_{t}\right]$. Instead of (30) and (31), we can use the strong Wolf condition, i.e., (relative) strong Wolfe stopping criterion.

$$
\begin{equation*}
\left|\phi^{\prime}\left(\rho_{t}\right)\right| \leq-c_{I I} \phi^{\prime}(0) \tag{32}
\end{equation*}
$$

where $c_{I I} \in(0,1)$ is a tuning parameter of the proposed quasi-Newton(L-BFGS) optimization for virtual convex loss. See also [19], [20] for related line search algorithms utilizing (32). In this article, for the strong-Wolfe stopping criterion (32), we create a new interval-based bisection line search(Algorithm 22). See [22], [41] for the various characteristics of the interval reduction method in general line search. The overall framework of quasi-Newton $(L$ BFGS) optimization for virtual convex loss is stated in Algorithm 1, which contains the interval-based bisection line search in Algorithm 2. See also Theorem IV.1 for the convergence of Algorithm 2,

```
Algorithm 1: Quasi-Newton optimization(L-BFGS) for virtual convex loss
    Input: \(h_{0}=\left(w_{0}, b_{0}\right)=0, \epsilon_{\text {tol } 1}=10^{-2}, \epsilon_{\text {tol } 2}=10^{-4}, c_{I I}=0.4\), MaxIter \(=100\).
    Output: \(h^{*}=\left(w^{*}, b^{*}\right)\).
    for \(0 \leq r \leq\) MaxIter do
        if \(\left\|\nabla \mathcal{F}\left(h_{t}\right)\right\|_{\infty} \leq \epsilon_{\text {tol } 1}\) or \(\left\|h_{t+1}-h_{t}\right\|_{\infty} \leq \epsilon_{t o l 2}\) then
            \(h^{*}=h_{t}\) and STOP
        Compute descent direction \(z_{t}\) by Quasi-Newton optimization(L-BFGS)
                    \(z_{t}=\operatorname{L-BFGS}\left(\nabla \mathcal{F}\left(h_{t}\right)\right)\)
```

        Compute step-Length \(\rho_{t}\) by Algorithm 2 with strong-Wolfe stopping criterion 32)
    8
            \(\rho_{r}=\underset{\rho \in \mathbb{R}_{\geq 0}}{\arg \min } \mathcal{F}\left(h_{t}+\rho z_{t}\right)\)
        Update \(h_{t+1}=h_{t}+\rho_{t} z_{t}\)
    \(h^{*}=h_{\text {MaxIter }+1}\)
    Theorem IV.1. Let $\phi$ be convex, differentiable, and bounded below. Then Algorithm 2 with an initial condition $\phi^{\prime}(0)<0$ converges to $\rho^{*}$ satisfying (32), where $c_{I I} \in(0,1)$, in finite steps.

Proof. Let us first consider the case that $\phi$ is a coercive function. Since $\phi^{\prime}(0)<0$ and $\phi^{\prime}$ is a non-decreasing function, there is $\rho_{\text {opt }}>0$ such that $\phi^{\prime}(\rho) \geq 0$ for all $\rho \geq \rho_{o p t}$. As noticed in line $6-7$ and line $10-11$ of Algorithm 2, there is $i$ th iteration such that $\phi^{\prime}\left(\rho_{i}\right)<\phi^{\prime}\left(\rho_{o p t}\right)=0<\phi^{\prime}\left(2 \rho_{i}\right)$. Therefore, the interval, which includes $\rho_{o p t}$, is established as $\left[\rho_{L}, \rho_{U}\right]=\left[\rho_{i}, 2 \rho_{i}\right]$. Then by the bisection algorithm in line $6-9$ and line $13,\left[\rho_{L}, \rho_{U}\right]$ is shrinking to $\rho_{o p t}$ and the strong-Wolfe stopping criterion in line 4 of Algorithm 2 is satisfied within finite steps. Now, we consider the case that $\phi$ is not a coercive function. Since $\phi$ is convex, bounded below, and $\phi^{\prime}(0)<0$, $\lim _{\rho \rightarrow+\infty} \phi^{\prime}(\rho) \rightarrow 0$ (line 11). Therefore, it stops by strong-Wolfe stopping criterion in line 4.

Remark IV.2. Besides Armijo (27) and Wolfe (28) criteria for line search, there is an additional criterion known as Goldstein condition [41]. By way of (29), it is reformulated as

$$
\begin{equation*}
\left(1-c_{I I I}\right) \phi^{\prime}(0) \leq \mathbf{E}_{\left[0, \rho_{t}\right]}\left(\phi^{\prime}\right) \leq c_{I I I} \phi^{\prime}(0) \tag{33}
\end{equation*}
$$

where $c_{I I I} \in(0,1 / 2)$ and $\phi^{\prime}(0)<0$. Unfortunately, this condition does not always include the solution of $\min _{\rho} \phi(\rho)$. To plug it into the quasi-Newton(L-BFGS) optimization for virtual loss, We need an additional curvature condition (31).

```
Algorithm 2: Interval-based bisection line search with (32)
    Input: \(c_{I I}=0.4\), itermax \(=100, \rho_{0}, \phi^{\prime}(\rho)=\left\langle\nabla \mathcal{F}\left(h_{t}+\rho z_{t}\right), z_{t}\right\rangle, \phi^{\prime}(0)<0,\left[\rho_{L}, \rho_{U}\right]=[0, \infty]\)
    Output: \(\rho^{*}\)
    for \(0 \leq i \leq\) itermax do
        if \(\left|\phi^{\prime}\left(\rho_{i}\right)\right| \leq-c_{I I} \phi^{\prime}(0)\) then
                \(\rho^{*}=\rho_{i}\) and STOP
        if \(0<\phi^{\prime}\left(\rho_{i}\right)\) then
            \(\rho_{U}=\rho_{i}\)
        else if \(\phi^{\prime}\left(\rho_{i}\right)<0\) then
            \(\rho_{L}=\rho_{i}\)
        if \(\rho_{U}=\infty\) then
            \(\rho_{i+1}=2 \rho_{L}\)
        else
            \(\rho_{i+1}=\frac{1}{2}\left(\rho_{L}+\rho_{U}\right)\)
    \(\rho^{*}=0\)
```


(a)

(b)

Fig. 5: Graphs of classification performance of the SIC model (15) with $\alpha=\alpha_{-}=\alpha_{+} \in[0,2]$. (a) Test accuracy(\%) vs. $\left|c_{\alpha}\right|$. The test accuracy is the average of all results obtained with $\alpha$ in (35). When $\left|c_{\alpha}\right|=2$, the best performance is achieved. (b) Test accuracy(\%) vs. $\alpha$. When $\frac{1}{5} \leq|\alpha-1| \leq 1$, the best performance is achieved with $\left|c_{\alpha}\right|=1$. However, when $|\alpha-1|<\frac{1}{5}$, the SIC model with $\left|c_{\alpha}\right| \geq 2$ shows better performance.

## V. Numerical experiments with the $20 \times 20$ SIC models

This Section reports the classification results acquired by the SIC model (15) and quasi-Newton(L-BFGS) for virtual convex loss(Algorithm 1 and 2]. We compare the proposed methodology with well-known classifiers: $\pi$ weighted convex focal loss [31], LIBLINEAR(logistic regression, SVM, and L2SVM) [12], [15], and LIBSVM(CSVC with RBF kernel) [6]. Note that Quasi-Newton(L-BFGS) for virtual convex loss is mainly implemented in MATLAB(version R2023b) based on [47]. This optimization algorithm is used for the SIC model (15) and the $\pi$-weighted convex focal loss [31]. LIBLINEAR(version 2.4.5) [15] and LIBSVM(version 3.3.2) [6] are mainly implemented in C/C++ language with MATLAB interface. All runs are performed on APPLE M2 Ultra with a 24core CPU and 192GB memory. The operating system is MacOS Sonoma(version 14.1). We use parfor in Matlab for parallel processing of all models, including LIBLINEAR and LIBSVM, in a 24-core CPU. In terms of multi-class datasets, the OVA(one-vs-all) strategy is used for all linear classification models. The OVO(one-vs-one) strategy is used for the kernel-based classification model LIBSVM [6].

Concerning quasi-Newton(L-BFGS) for virtual convex loss, as observed in Figure 4, it is recommended to select $m \in[20,50]$ for two-loop iterations of L-BFGS and $c_{I I} \in[0.1,0.5]$ for the interval-based bisection line search(Algorithm 22). We choose $m=40$ and $c_{I I}=0.4$ considering performance-computation complexity.

For stopping criterions of quasi-Newton(L-BFGS) for virtual convex loss, we use $\left\|\nabla \mathcal{F}\left(h_{t}\right)\right\|_{\infty} \leq \epsilon_{\text {tol } 1}$ and $\left\|h_{t+1}-h_{t}\right\|_{\infty} \leq \epsilon_{\text {tol } 2}$ where $\epsilon_{\text {tol } 1}=10^{-2}$ and $\epsilon_{\text {tol } 2}=10^{-4}$ (Algorithm 11. We could select a smaller $c_{I I}$ for exact line search, used in other quasi-Newton optimization, such as nonlinear conjugate gradient [19].
In order to use the $\pi$-weighted convex focal loss [31] discussed in Remark III.9, we need to set three parameters: $\gamma, \xi$, and $\pi$. Following the recommendations in [31], we choose $\gamma=1,2,3,4$ and $\xi=0,1$. As for $\pi$, we select 19 regular points ranging from 0.05 to 0.95 . This gives us 152 convex focal losses, expressed as a $(\pi, \gamma: \xi)$-matrix.

We have selected LIBLINEAR [12] and LIBSVM [6] as our standard for balanced linear classification models and non-linear classification models, respectively. For logistic regression, we use logistic loss, hinge-loss for SVM, and squared hinge-loss for L2SVM. To learn an inhomogeneous hyperplane, we set $B=1$. We use the primal formulation $(s=0)$ for logistic regression and the dual formulation $(s=3)$ for SVM. As for L2SVM, we use the primal formulation $(s=2)$. In LIBSVM [6], we use C-SVC(support vector classification) $(s=0)$ with the RBF kernel $K\left(x_{i}, x_{j}\right)=\exp \left(-\nu\left\|x_{i}-x_{j}\right\|^{2}\right)(t=2)$.

All models have an $\ell_{2}$-regularizer $\frac{\lambda}{2}\|w\|^{2}$. In terms of regularization parameter $\lambda$ for the cost-sensitive learning framework (4), including $20 \times 20$ SIC models (15) and $19 \times 8 \pi$-weighted convex focal loss models in Remark [III.9, we use CV(cross-validation) with candidates in (34) as recommended in LIBSVM [6].

$$
\begin{equation*}
\lambda=2^{r}, r=-14,-13,-12, \cdots, 5 \tag{34}
\end{equation*}
$$

In LIBLINEAR and LIBSVM, the regularization parameter $\lambda$ is located on the loss function. Therefore, we use $C=\lambda^{-1}$ with (34) for CV. For LIBSVM, in addition to the regularization parameter $C$ on the loss function, the RBF kernel parameter $\nu$ is cross-validated with candidates $\nu=2^{r}$ and $r=-14, . ., 5$.

Regarding benchmark datasets [8], they are pre-processed and normalized in each feature dimension with mean zero and variance one [23], except for when the variance of the raw data is zero. This process reduces the effect of scale imbalance of datasets. The scale-class-imbalance ratio $r_{s c}$ (1) of two-class and multi-class datasets is presented in Table $V$ and Table VII, respectively. In the case of two-class datasets in Table $V$, the mean value of $r_{s c}$ of training dataset is $\mathbf{E}\left(r_{s c} \mathbf{T}\right) \approx 1.61$. Also, we have $\min r_{s c} \mathbf{T}=0.49$ and $\max r_{s c} \mathbf{T}=7$. Thus, the two-class datasets used in our experiments are roughly well-balanced. However, most of the two-class datasets have variations between $r_{s c} \mathbf{T}$ and $r_{s c}$ of test dataset $\left(r_{s c} \mathbf{T e}\right)$. The raw format of each benchmark dataset is available in the UCI machine learning repository [52]. As commented in [55], we reorganize datasets in [8]. Each dataset is separated into the non-overlapped training and test datasets. The training dataset of each dataset is randomly shuffled for 4 -fold CV [6]. Table $\nabla(51$ two-class datasets) and Table VI 67 multi-class datasets) include all information of datasets such as number of instances, size of training dataset, size of test dataset, feature dimension, number of classes, classimbalance ratio $r_{c}$ for combined/training/test dataset, and scale-class-imbalance ratio $r_{s c}$ for combined/training/test dataset. The experiments are conducted five times using randomly selected CV datasets, with a fixed initial condition of $\left(w_{0}, b_{0}\right)=(0,0)$. For $\alpha$ and $c_{\alpha}$ of SIC model (15), we conducted a preliminary experiment with the reduced class of SIC model $\left(\alpha=\alpha_{+}=\alpha_{-}\right)$. We found that the best test classification accuracy is obtainable when $\left|c_{\alpha}\right|=2$. For general purposes, $\left|c_{\alpha}\right| \in[1,10]$ is a possible choice. When $\alpha$ is not close to 1 , the SIC model with $\left|c_{\alpha}\right|=1$ shows the best performance. The detailed information is provided in Figure 5. For the experiments in this Section, we set $c_{\alpha}=2$ for $\alpha>1$ and $c_{\alpha}=-2$ for $\alpha<1$. Thus, $\alpha_{ \pm}$are the only tuning parameters for which we use the following 20 different values in $[0,2]$ :

$$
\begin{equation*}
\alpha_{ \pm} \in\left\{\frac{0}{1}, \frac{1}{2}, \frac{2}{3}, \cdots, \frac{9}{10}, \frac{11}{10}, \frac{10}{9}, \frac{9}{8}, \cdots, \frac{3}{2}, \frac{2}{1}\right\} \tag{35}
\end{equation*}
$$

This gives us $20 \times 20$ SIC models. The characteristic of each dataset could be captured by the large class of hyperplanes $h_{\left(\alpha_{+}, \alpha_{-}\right)}^{*}=\left(w_{\left(\alpha_{+}, \alpha_{-}\right)}^{*}, b_{\left(\alpha_{+}, \alpha_{-}\right)}^{*}\right)$ learned via the $20 \times 20$ SIC models (15), as noticed in Theorem III. 6 and Example III.8. The details are as follows.

## A. Performance evaluation of $20 \times 20$ SIC models

Table II summarizes the classification accuracy (\%) and computation time of all experiments conducted on 118 datasets. The acronym TOP1 refers to a group of SIC models that have the highest test accuracy for each dataset, while MaxA/Max2/MaxM refers to an SIC model with the best test accuracy for all-, two-, and multiclass datasets. The same notations are used for $\pi$-weighted convex focal loss: TOP1-FL, MaxA-FL, Max2-FL, and

| MODEL | SIC model 15 |  |  |  | Convex Focal Loss 31 |  |  |  | LIBLINEAR 12] |  |  | LIBSVM 6 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| SubModel $\left(\alpha_{+}, \alpha_{-}\right) \text {or }(\pi, \gamma, \xi)$ | TOP1 | $\begin{gathered} \text { MaxA } \\ \left(\frac{7}{8}, \frac{8}{7}\right) \\ \hline \end{gathered}$ | $\begin{gathered} \text { Max2 } \\ \left(\frac{3}{4}, 0\right) \\ \hline \end{gathered}$ | $\begin{gathered} \text { MaxM } \\ \left(\frac{8}{9}, \frac{11}{10}\right) \\ \hline \end{gathered}$ | TOP1-FL | $\begin{gathered} \text { MaxA-FL } \\ (0.5,2,1) \\ \hline \end{gathered}$ | $\begin{gathered} \text { Max2-FL } \\ (0.5,3,0) \\ \hline \end{gathered}$ | $\begin{gathered} \hline \text { MaxM-FL } \\ (0.6,2,1) \\ \hline \end{gathered}$ | Logistic (primal) | $\begin{aligned} & \hline \begin{array}{l} \text { SVM } \\ \text { (dual) } \end{array} \end{aligned}$ | $\begin{aligned} & \text { L2SVM } \\ & \text { (primal) } \end{aligned}$ | $\begin{gathered} \text { C-SVC } \\ \text { RBF Kernel } \end{gathered}$ |
| Mean acc(\%) of Two Class | 83.96 | 82.49 | 82.51 | 82.36 | 83.80 | 82.14 | 82.37 | 81.39 | 82.11 | 81.59 | 82.06 | 83.22 |
| Mean acc(\%) of Multi Class | 77.30 | 75.57 | 75.19 | 75.57 | 76.68 | 75.53 | 75.36 | 75.55 | 74.75 | 72.94 | 74.18 | 79.96 |
| Mean acc(\%) of All Class | 80.18 | 78.56 | 78.35 | 78.50 | 79.76 | 78.39 | 78.39 | 78.07 | 77.93 | 76.68 | 77.58 | 81.37 |
| Time of all class | 423 m | 106s | 122s | 98s | 81m | 88s | 88s | 88s | 60s | 109s | 57s | 2077m |

TABLE II: A comparison of the SIC model (15] with benchmark models: $\pi$-weighted convex focal loss [31], LIBLINEAR [12], and LIBSVM [6]. In this comparison, TOP1 refers to a group of SIC models with the best test accuracy for each dataset, while MaxA/Max2/MaxM is an SIC model with the best test accuracy for all-, two-, and multi-class. The same notations are used for $\pi$-weighted convex focal loss: TOP1-FL, MaxA-FL, Max2-FL, and MaxM-FL. In the two-class problems, TOP1 performs better than kernel-based LIBSVM and TOP1-FL. Max2 of the SIC model offers the best test classification accuracy among various linear classifiers. In the multi-class problems, although the SIC model's TOP1 accuracy is less than kernel-based LIBSVM, it still achieves $77.30 \%$ accuracy, which is $0.62 \%$ better than TOP1-FL. When the models' parameters are fixed, the SIC model performs similarly to the convex focal loss having the external $\pi$-weight parameter.

MaxM-FL. The test classification accuracy of each dataset is reported in Table $[$ III for two class datasets and in Table IV for multi-class datasets. Note that $\operatorname{MaxA}\left(\alpha_{+}=\frac{7}{8}, \alpha_{-}=\frac{8}{7}\right)$ achieves $78.56 \%$. On the other hand, MaxA$\mathrm{FL}(\pi=0.5, \gamma=2, \xi=1)$ obtains $78.39 \%$. Over half of all SIC models obtain at least $78.20 \%$ accuracy. Out of all the $\pi$-weighted convex focal losses, only $10 \%$ can achieve the same level of accuracy as the proposed SIC model. This implies that the SIC model is less sensitive to the parameter than the $\pi$-weighted convex focal loss. Therefore, the SIC model could serve as an alternative cost-sensitive learning framework without external $\pi$-weight. Refer to Figure 9 for additional information. The details are as follows.

In the case of two-class, of which the training dataset is close to the well-balanced condition, TOP1 achieves the best results, i.e., $0.74 \%$ better than the kernel-based classifier LIBSVM(C-SVC with RBF kernel) and $0.16 \%$ better than TOP1-FL. When the parameters of the SIC model are fixed, its performance is still better than other linear classifiers, such as $\pi$-weighted convex focal loss and LIBLINEAR. For instance, $\operatorname{Max} 2\left(\alpha_{+}=\frac{3}{4}, \alpha_{-}=0\right)$ has $82.51 \%$ accuracy, which is $0.14 \%$ better than Max2-FL and $0.4 \%$ better than logistic regression, the best model of LIBLINEAR. As shown in Figure 6(a), the test accuracy of all SIC models is in the range of [80.64\%, 82.51\%]. More than $35 \%$ of all SIC models achieve at least $82.20 \%$ test accuracy. On the other hand, the test accuracy of all convex focal losses is in the range of $[65.93 \%, 82.37 \%]$. Out of all the convex focal loss, only $2 \%$ can achieve $82.20 \%$ test accuracy. It appears that the SIC models are quite resilient to internal parameter changes. Specifically, Figure 7 (a) shows an $\boldsymbol{X}$-shaped pattern. This pattern covers a much larger area compared to the best test accuracy area of convex focal losses in Figure 7 (c). The $\boldsymbol{X}$-shaped pattern relates to the pattern of $\eta$ in Figure 3 (d). It represents a small deviation from the balanced SIC model, which has $\left|\alpha_{+}-1\right|=\left|\alpha_{-}-1\right|$. Essentially, the virtual SIGTRON-induced loss functions $L_{\alpha_{+}, c_{+}}^{S}$ and $L_{\alpha_{-}, c_{-}}^{S}$ of the SIC model have similar polynomial orders, i.e., $k_{+} \approx k_{-}$. Figure 8 (b) horse-colic demonstrates the $\boldsymbol{X}$-shaped pattern.

It is important to note that spectf dataset in Table $\nabla$ is a typical $r_{s c}$-inconsistent dataset. By using this dataset, the connection between $\eta=\frac{k_{-}}{k_{-}+k_{+}}$and the movement of the hyperplane $h_{\alpha_{+}, \alpha_{-}}^{*}(x)=0$ is empirically demonstrated in Figure 3. Notably, the best test accuracy of the dataset is observed in the region $\left(\alpha_{+}, \alpha_{-}\right)=(-, 2)$, which is outside the $\boldsymbol{X}$-shaped pattern.
Regarding multi-class datasets, the kernel-based classifier LIBSVM(C-SVC with RBF kernel) achieves the highest test classification accuracy. As presented in Table ח. although the test accuracy of TOP1 is less than kernel-based LIBSVM, it still achieves a respectable $77.30 \%$, which is $0.62 \%$ better than TOP1-FL. In Figure 6(b), we observe that the SIC model, which has only internal polynomial order parameters $k_{ \pm}=\frac{1}{\alpha_{ \pm}-1 \mid}$, performs similarly to the convex focal loss, which has the external $\pi$-weight parameter and the internal $\xi$ and $\gamma$ parameters. Note that Figure 7 (b) shows a pattern of the best-performing SIC model in the ( $\alpha_{+}, \alpha_{-}$)-matrix. Compared to two-class, the $\boldsymbol{X}$-shaped pattern is rounded and biased toward $\alpha_{-}>1$. In the case of convex focal loss in Figure 7 (d), the best-performing region is much larger than the two-class convex focal loss. The region is shifted towards $\pi<0.5$.

Lastly, regarding computation time, L2SVM(primal) and logistic regression(primal) of LIBLINEAR are the fastest models. These models use the truncated Newton method [15] that is based on the unique Hessian structure of the large-margin linear classifier. On the other hand, for both the SIC model and convex focal loss, the proposed


Fig. 6: The statistical distributions of $20 \times 20 \mathrm{SIC}$ models and $19 \times 8$ convex focal losses with respect to test classification accuracy. (a) The two-class histogram. The SIC models are in the [ $80.64 \%, 82.51 \%$ ] range, and convex focal losses are in the $[65.93 \%, 82.37 \%]$ range. The SIC model is less sensitive to internal parameters $\alpha_{ \pm}$ and outperforms the $\pi$-weighted convex focal losses. (b) The multi-class histogram. The SIC models are in the $[73.64 \%, 75.57 \%]$ range, and convex focal losses are in the $[74.08 \%, 75.55 \%]$ range. Although the SIC models have no external $\pi$-weight parameters, they show comparable performance to $\pi$-weighted convex focal loss. For more information on the matrix pattern of the models, refer to Figure 7 .


Fig. 7: Test accuracy(\%) patterns of $20 \times 20$ SIC models for (a) two-class and (b) multi-class. Test accuracy(\%) patterns of $19 \times 8$ convex focal losses for (c) two-class and (d) multi-class. As shown in (a), SIC models exhibit an $\boldsymbol{X}$-shaped pattern, i.e., $\left|\alpha_{-}-1\right| \approx\left|\alpha_{+}-1\right|$. This indicates that the best-performing SIC models have similar polynomial order between $L_{\alpha_{+}, c_{+}}^{S}$ and $L_{\alpha_{-}, c_{-}}^{S}$. This $\boldsymbol{X}$-shaped pattern covers a much larger region than the $\pi$ weighted convex focal losses in (c). For more information and statistical distribution of the models, refer to Figure 6.

Quasi-Newton(L-BFGS) optimization for virtual convex loss is used. As shown in Figure 9 (b) in Appendix B, the $\pi$-weighted convex focal loss with $\pi=0.5, \gamma=1, \xi=0$, which corresponds to the logistic loss of LIBLINEAR, achieves reasonable performance-computation complexity, resulting in $78.30 \%$ test accuracy at 83 seconds. It is worth noting that the logistic regression of LIBLINEAR only obtains $77.93 \%$ test accuracy at 60 seconds.

Figure 8 demonstrates patterns of test classification accuracy for two-class datasets, statlog-australian-credit and horse-colic and for multi-class datasets, ecoli, arrhythmia, energy-yl, and energy-y2. Overall, the best test accuracy


Fig. 8: Matrix patterns of the $20 \times 20 \mathrm{SIC}$ models of the test classification accuracy of (a) statlog-australiancredit, (b) horse-colic, (c) ecoli, (d) arrhythmia, (e) energy-y1, and (f) energy-y2. Here statlog-australian-credit and horse-colic are two-class datasets. And ecoli, arrhythmia, energy-y1, and energy-y2 are multi-class datasets. Both energy-yl and energy-y2 have the same input dataset but look for hyperplanes for opposite outputs, i.e., yl for the heating load, while $y 2$ for the cooling load [51]. The best performing ( $\alpha_{+}, \alpha_{-}$) for energy-yl and energy-y2 exhibit opposite patterns: $\left(\alpha_{+}, \alpha_{-}\right) \approx(2,-)$ for energy-yl, and $\left(\alpha_{+}, \alpha_{-}\right) \approx(-, 2)$ for energy-y2. Overall, the best-performing region of multi-class datasets is more localized than that of two-class datasets.
regions of multi-class datasets are more localized than those of two-class datasets. The $\boldsymbol{X}$-shaped pattern in Figure 7 (a) is also observed in the horse-colic dataset in Figure 8 (b). Both energy-yl and energy-y2 have the same input dataset but look for hyperplanes for opposite outputs. Specifically, energy-yl is used to determine the heating load, while energy-y2 is used to determine the cooling load [51]. The best performing ( $\alpha_{+}, \alpha_{-}$) for energy-yl and energy-y2 exhibit opposite patterns: $\left(\alpha_{+}, \alpha_{-}\right) \approx(2,-)$ for energy-yl, and $\left(\alpha_{+}, \alpha_{-}\right) \approx(-, 2)$ for energy-y2. Refer to Figure 8 (e) and (f) for further details. Understanding the correlation between the pattern of ( $\alpha_{+}, \alpha_{-}$)-matrix and the structure of each dataset can be a valuable tool for multi-label classification and imbalanced classification.

| MODEL | SIC model 15 |  |  |  | Convex Focal Loss 31 |  |  |  | LIBLINEAR 12] |  |  | LIBSVM 6 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| SubModel $\left(\alpha_{+}, \alpha_{-}\right) \text {or }(\pi, \gamma, \xi)$ | TOP1 | $\begin{gathered} \text { MaxA } \\ \left(\frac{7}{8}, \frac{8}{7}\right) \\ \hline \end{gathered}$ | $\begin{gathered} \text { Max2 } \\ \left(\frac{3}{4}, 0\right) \\ \hline \end{gathered}$ | $\begin{gathered} \text { MaxM } \\ \left(\frac{8}{9}, \frac{11}{10}\right) \\ \hline \end{gathered}$ | TOP1-FL | $\begin{gathered} \hline \text { MaxA-FL } \\ (0.5,2,1) \\ \hline \end{gathered}$ | $\begin{gathered} \hline \text { Max2-FL } \\ (0.5,3,0) \\ \hline \end{gathered}$ | $\begin{gathered} \hline \text { MaxM-FL } \\ (0.6,2,1) \\ \hline \end{gathered}$ | Logistic (primal) | $\begin{aligned} & \hline \text { SVM } \\ & \text { (dual) } \end{aligned}$ | $\begin{gathered} \hline \text { L2SVM } \\ \text { (primal) } \\ \hline \end{gathered}$ | $\begin{gathered} \text { C-SVC } \\ \text { RBF Kernel } \end{gathered}$ |
| acute-inflammation | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 |
| acute-nephritis | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 | 100.00 |
| adult | 84.31 | 84.24 | 84.00 | 84.26 | 84.38 | 84.38 | 84.29 | 83.73 | 84.28 | 84.33 | 84.05 | 85.03 |
| balloons | 87.50 | 87.50 | 87.50 | 87.50 | 87.50 | 87.50 | 87.50 | 72.50 | 87.50 | 87.50 | 87.50 | 87.50 |
| bank | 89.00 | 88.79 | 88.82 | 88.81 | 89.51 | 88.75 | 88.81 | 88.68 | 88.83 | 88.19 | 88.81 | 88.75 |
| blood | 77.17 | 76.58 | 76.68 | 76.15 | 76.74 | 75.99 | 76.20 | 76.74 | 76.20 | 76.20 | 75.67 | 76.84 |
| breast-cancer | 72.73 | 72.03 | 71.61 | 71.89 | 73.29 | 71.61 | 71.47 | 72.45 | 71.05 | 69.51 | 70.77 | 75.66 |
| breast-cancer-wisc | 96.85 | 96.62 | 96.56 | 96.62 | 96.85 | 96.50 | 96.33 | 95.82 | 96.50 | 96.62 | 96.68 | 95.99 |
| breast-cancer-wisc-diag | 98.38 | 98.17 | 97.75 | 98.10 | 98.31 | 98.24 | 98.17 | 98.24 | 98.24 | 97.46 | 97.96 | 98.38 |
| breast-cancer-wisc-prog | 81.41 | 79.60 | 78.59 | 79.80 | 82.02 | 78.79 | 79.19 | 79.60 | 78.38 | 75.15 | 79.39 | 78.38 |
| chess-krvkp | 96.93 | 96.12 | 96.77 | 96.46 | 97.01 | 96.71 | 96.53 | 96.78 | 96.48 | 96.33 | 96.68 | 98.82 |
| congressional-voting | 60.92 | 58.06 | 59.17 | 58.80 | 61.94 | 59.26 | 58.53 | 60.46 | 57.70 | 59.91 | 57.70 | 57.33 |
| conn-bench-sonar-mines-rocks | 79.23 | 76.35 | 75.96 | 77.88 | 78.46 | 75.77 | 77.12 | 75.19 | 75.58 | 77.50 | 75.77 | 84.42 |
| connect-4 | 75.49 | 75.45 | 75.39 | 75.45 | 75.50 | 75.42 | 75.47 | 75.38 | 75.47 | 75.38 | 75.41 | 86.26 |
| credit-approval | 89.39 | 88.58 | 87.48 | 89.16 | 89.62 | 88.29 | 88.75 | 87.36 | 88.58 | 87.54 | 87.88 | 86.84 |
| cylinder-bands | 74.84 | 73.91 | 73.28 | 73.67 | 74.92 | 72.42 | 73.12 | 71.41 | 73.83 | 74.14 | 73.52 | 77.34 |
| echocardiogram | 86.77 | 84.92 | 84.92 | 84.62 | 87.38 | 84.62 | 84.62 | 86.15 | 85.85 | 87.69 | 86.15 | 87.38 |
| fertility | 89.60 | 88.40 | 88.00 | 89.60 | 90.00 | 84.80 | 89.60 | 86.80 | 85.60 | 87.20 | 86.00 | 86.80 |
| haberman-survival | 74.12 | 73.59 | 73.59 | 73.59 | 74.38 | 73.59 | 73.59 | 72.55 | 73.86 | 74.77 | 73.86 | 71.63 |
| heart-hungarian | 87.89 | 86.67 | 86.67 | 86.67 | 88.03 | 85.99 | 86.53 | 84.90 | 86.67 | 85.03 | 86.67 | 86.12 |
| hepatitis | 81.04 | 79.22 | 77.92 | 77.66 | 81.82 | 75.84 | 77.40 | 76.88 | 77.66 | 75.06 | 76.62 | 81.04 |
| hill-valley | 92.08 | 83.43 | 92.08 | 81.39 | 86.34 | 81.06 | 84.75 | 76.63 | 80.86 | 56.70 | 80.00 | 65.41 |
| horse-colic | 89.71 | 88.82 | 87.65 | 87.35 | 89.71 | 88.24 | 89.41 | 87.65 | 87.65 | 87.06 | 87.06 | 85.00 |
| ilpd-indian-liver | 73.40 | 73.40 | 72.10 | 72.16 | 72.58 | 72.23 | 71.96 | 71.55 | 71.96 | 71.48 | 73.06 | 71.41 |
| ionosphere | 88.80 | 86.29 | 86.40 | 86.51 | 87.09 | 86.63 | 86.63 | 86.97 | 88.34 | 87.77 | 86.74 | 94.63 |
| magic | 79.64 | 79.12 | 79.64 | 79.01 | 79.45 | 79.08 | 79.15 | 78.20 | 79.04 | 78.85 | 78.97 | 87.20 |
| miniboone | 90.35 | 90.16 | 89.77 | 90.26 | 90.30 | 90.30 | 90.26 | 90.14 | 89.79 | 89.98 | 89.04 | 93.50 |
| molec-biol-promoter | 84.53 | 77.36 | 76.60 | 76.60 | 79.25 | 76.98 | 77.36 | 76.60 | 78.11 | 76.23 | 76.23 | 76.98 |
| mammographic | 83.75 | 83.25 | 82.96 | 82.75 | 83.58 | 82.87 | 83.46 | 83.46 | 83.50 | 83.17 | 83.08 | 83.21 |
| mushroom | 95.24 | 94.52 | 95.02 | 94.60 | 97.46 | 95.62 | 94.58 | 94.85 | 94.40 | 97.64 | 93.89 | 100.00 |
| musk-1 | 84.45 | 81.93 | 83.28 | 82.86 | 85.46 | 82.69 | 82.18 | 84.87 | 81.51 | 83.28 | 83.19 | 91.18 |
| musk-2 | 94.99 | 94.63 | 94.56 | 94.76 | 95.25 | 94.80 | 94.63 | 95.06 | 94.67 | 95.02 | 94.72 | 99.19 |
| oocytes-merluccius-nucleus-4d | 83.01 | 82.04 | 82.04 | 81.41 | 82.54 | 82.07 | 82.11 | 81.57 | 81.80 | 80.74 | 82.74 | 82.07 |
| oocytes-trisopterus-nucleus-2f | 81.32 | 79.91 | 80.39 | 79.08 | 80.88 | 80.39 | 78.55 | 78.42 | 78.51 | 78.68 | 78.73 | 82.19 |
| ozone | 97.22 | 97.13 | 97.13 | 97.08 | 97.32 | 97.07 | 97.11 | 97.08 | 97.15 | 97.10 | 97.13 | 96.99 |
| parkinsons | 84.12 | 82.68 | 82.47 | 83.09 | 85.77 | 83.92 | 83.09 | 82.06 | 82.47 | 83.51 | 83.71 | 91.34 |
| pima | 76.98 | 75.52 | 75.36 | 76.61 | 77.14 | 76.72 | 76.61 | 76.09 | 76.35 | 75.31 | 76.35 | 73.80 |
| pittsburg-bridges-T-OR-D | 89.02 | 87.84 | 88.24 | 88.63 | 90.20 | 88.63 | 88.24 | 87.84 | 89.80 | 86.67 | 90.20 | 87.06 |
| planning | 70.55 | 67.47 | 65.27 | 67.03 | 71.43 | 65.93 | 67.47 | 67.91 | 64.40 | 70.11 | 65.27 | 69.45 |
| ringnorm | 77.95 | 77.37 | 77.86 | 77.11 | 77.87 | 77.25 | 76.89 | 74.16 | 76.86 | 77.51 | 77.11 | 98.74 |
| spambase | 92.88 | 92.82 | 92.72 | 92.44 | 93.03 | 92.77 | 92.45 | 91.42 | 92.37 | 92.78 | 92.22 | 93.23 |
| spect | 66.67 | 62.90 | 62.90 | 62.26 | 67.85 | 61.83 | 62.04 | 59.78 | 64.30 | 65.16 | 62.37 | 60.00 |
| spectf | 64.60 | 56.36 | 57.86 | 52.94 | 61.50 | 50.48 | 52.09 | 46.31 | 48.98 | 47.27 | 49.30 | 46.84 |
| statlog-australian-credit | 68.12 | 67.13 | 67.71 | 66.90 | 67.94 | 67.54 | 67.13 | 62.55 | 67.13 | 67.83 | 66.96 | 66.26 |
| statlog-german-credit | 77.16 | 76.96 | 76.44 | 76.60 | 77.40 | 75.92 | 76.20 | 74.96 | 76.80 | 76.24 | 77.16 | 76.08 |
| statlog-heart | 88.59 | 87.11 | 86.52 | 88.00 | 87.56 | 87.41 | 87.26 | 84.15 | 86.67 | 84.59 | 87.70 | 84.74 |
| tic-tac-toe | 97.91 | 97.91 | 97.91 | 97.91 | 97.91 | 97.91 | 97.91 | 97.91 | 97.91 | 97.91 | 97.91 | 98.62 |
| titanic | 78.09 | 77.55 | 77.55 | 77.55 | 78.09 | 77.55 | 77.55 | 77.55 | 77.55 | 77.55 | 77.55 | 78.42 |
| trains | 64.00 | 60.00 | 60.00 | 60.00 | 60.00 | 60.00 | 60.00 | 60.00 | 60.00 | 60.00 | 60.00 | 40.00 |
| twonorm | 97.66 | 97.57 | 97.54 | 97.56 | 97.68 | 97.62 | 97.44 | 97.40 | 97.57 | 97.49 | 97.49 | 97.69 |
| vertebral-column-2clases | 85.68 | 83.23 | 81.29 | 82.97 | 87.61 | 83.35 | 83.10 | 86.06 | 82.97 | 81.94 | 82.32 | 82.32 |
| Mean | 83.96 | 82.49 | 82.51 | 82.36 | 83.80 | 82.14 | 82.37 | 81.39 | 82.11 | 81.59 | 82.06 | 83.22 |

TABLE III: A comparison of two-class test classification accuracy of the SIC model 15) to $\pi$-weighted convex focal loss [31], LIBLINEAR [12], and LIBSVM [6]. Our results indicate that TOP1 performs better than kernelbased LIBSVM and the corresponding TOP1-FL. Max2 offers the best test classification accuracy among various linear classifiers.

## VI. Conclusion

This article introduces SIGTRON, an extended asymmetric sigmoid function with Perceptron, and its virtualized loss function called virtual SIGTRON-induced loss function. Based on this loss function, we propose the SIGTRONimbalanced classification (SIC) model for cost-sensitive learning. Unlike other models, the SIC model does not use an external $\pi$-weight on the loss function but instead has an internal two-dimensional parameter $\left(\alpha_{+}, \alpha_{-}\right)$-matrix. We show that when a training dataset is close to a well-balanced condition, the SIC model is moderately resilient to variations in the dataset through the skewed hyperplane equation. When $r_{s c}$ is not severe, the proposed SIC model could be used as an alternative cost-sensitive learning model that does not require an external $\pi$-weight parameter. Additionally, we introduce quasi-Newton(L-BFGS) optimization for virtual convex loss with an intervalbased bisection line search. This optimization is a competitive framework for a convex minimization problem, compared to conventional L-BFGS with cubic-interpolation-based line search. We utilize the proposed optimization framework for the SIC model and the $\pi$-weighted convex focal loss. Our SIC model has shown better performance in terms of test classification accuracy with 118 diverse datasets compared to the $\pi$-weighted convex focal loss and LIBLINEAR. In binary classification problems, where the severity of $r_{s c}$ is not high, selecting the best SIC

| MODEL | SIC model [15] |  |  |  | Convex Focal Loss 31$]$ |  |  |  | LIBLINEAR 12 |  |  | LIBSVM 6 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| SubModel $\left(\alpha_{+}, \alpha_{-}\right) \text {or }(\pi, \gamma, \xi)$ | TOP1 | $\begin{gathered} \text { MaxA } \\ \left(\frac{7}{8}, \frac{8}{7}\right) \\ \hline \end{gathered}$ | $\begin{gathered} \text { Max2 } \\ \left(\frac{3}{4}, 0\right) \\ \hline \end{gathered}$ | $\begin{gathered} \text { MaxM } \\ \left(\frac{8}{9}, \frac{11}{10}\right) \\ \hline \end{gathered}$ | $\begin{gathered} \text { TOP1-FL } \\ - \\ \hline \end{gathered}$ | $\begin{gathered} \text { MaxA-FL } \\ (0.5,2,1) \\ \hline \end{gathered}$ | $\begin{gathered} \text { Max2-FL } \\ (0.5,3,0) \\ \hline \end{gathered}$ | $\begin{gathered} \text { MaxM-FL } \\ (0.6,2,1) \\ \hline \end{gathered}$ | Logistic (primal) | $\begin{aligned} & \text { SVM } \\ & \text { (dual) } \end{aligned}$ | $\begin{aligned} & \text { L2SVM } \\ & \text { (primal) } \end{aligned}$ | C-SVC RBF Kernel |
| abalone | 65.78 | 65.23 | 65.43 | 65.08 | 65.57 | 65.31 | 65.12 | 65.36 | 65.18 | 60.31 | 65.14 | 66.22 |
| annealing | 87.77 | 86.27 | 87.02 | 86.12 | 87.12 | 86.22 | 85.91 | 86.97 | 86.87 | 87.77 | 86.97 | 92.08 |
| arrhythmia | 69.73 | 65.40 | 68.85 | 67.88 | 69.47 | 68.32 | 68.76 | 67.88 | 68.50 | 65.93 | 64.16 | 67.52 |
| audiology-std | 80.00 | 76.00 | 78.40 | 76.00 | 80.00 | 72.80 | 71.20 | 73.60 | 70.40 | 68.80 | 68.80 | 65.60 |
| balance-scale | 88.46 | 88.14 | 88.01 | 88.27 | 88.46 | 88.40 | 88.14 | 88.40 | 88.21 | 88.21 | 88.08 | 98.85 |
| breast-tissue | 69.06 | 66.04 | 65.28 | 66.04 | 66.42 | 66.42 | 66.42 | 66.42 | 65.28 | 64.15 | 66.04 | 65.66 |
| car | 82.64 | 82.52 | 81.48 | 82.41 | 82.94 | 82.36 | 82.36 | 82.04 | 82.41 | 79.93 | 81.34 | 98.47 |
| cardiotocography-10clases | 79.49 | 78.46 | 77.57 | 78.19 | 78.74 | 77.46 | 77.99 | 77.84 | 78.04 | 73.89 | 77.38 | 80.94 |
| cardiotocography-3clases | 90.05 | 89.73 | 89.56 | 89.76 | 90.48 | 89.78 | 89.71 | 89.50 | 89.84 | 90.08 | 89.76 | 91.95 |
| chess-krvk | 28.07 | 27.83 | 27.38 | 27.77 | 28.10 | 28.03 | 27.82 | 27.96 | 27.84 | 17.36 | 27.59 | 76.09 |
| conn-bench-vowel-deterding | 57.65 | 52.73 | 54.24 | 52.80 | 55.68 | 53.26 | 52.65 | 54.09 | 51.89 | 50.53 | 54.47 | 96.06 |
| contrac | 51.66 | 50.98 | 50.65 | 50.82 | 51.22 | 50.38 | 50.87 | 50.49 | 50.87 | 46.44 | 50.03 | 52.47 |
| dermatology | 99.34 | 97.81 | 97.38 | 98.03 | 98.14 | 97.60 | 98.03 | 98.03 | 97.81 | 96.72 | 97.27 | 98.14 |
| ecoli | 88.93 | 87.50 | 88.21 | 87.74 | 88.57 | 88.10 | 88.21 | 88.21 | 87.98 | 86.07 | 87.98 | 85.60 |
| energy-y1 | 88.85 | 85.26 | 85.47 | 85.57 | 89.06 | 85.00 | 85.57 | 85.89 | 86.20 | 85.16 | 86.09 | 94.06 |
| energy-y2 | 91.15 | 89.48 | 88.59 | 88.85 | 90.10 | 88.75 | 88.23 | 88.54 | 89.43 | 88.85 | 89.69 | 92.71 |
| flags | 54.85 | 53.40 | 53.81 | 53.40 | 54.23 | 52.99 | 53.20 | 53.20 | 52.99 | 55.05 | 53.40 | 53.61 |
| glass | 68.22 | 62.06 | 63.93 | 62.80 | 65.42 | 64.86 | 62.80 | 64.11 | 63.93 | 60.00 | 65.23 | 70.09 |
| hayes-roth | 62.14 | 53.57 | 53.57 | 53.57 | 57.86 | 53.57 | 53.57 | 53.57 | 50.71 | 42.14 | 45.00 | 77.14 |
| heart-cleveland | 64.11 | 61.99 | 62.25 | 62.52 | 63.84 | 62.65 | 62.65 | 61.85 | 61.32 | 61.19 | 62.78 | 59.60 |
| heart-switzerland | 41.97 | 39.34 | 39.02 | 39.34 | 39.67 | 37.05 | 38.03 | 38.03 | 35.74 | 37.38 | 35.74 | 40.33 |
| heart-va | 32.20 | 29.40 | 29.40 | 29.60 | 32.20 | 29.60 | 29.40 | 29.80 | 28.20 | 31.60 | 27.00 | 28.00 |
| image-segmentation | 91.14 | 90.13 | 90.70 | 90.67 | 91.30 | 91.28 | 90.55 | 90.89 | 90.39 | 90.82 | 90.18 | 90.94 |
| iris | 97.33 | 93.33 | 94.67 | 94.40 | 96.00 | 94.93 | 94.67 | 94.67 | 94.67 | 91.47 | 94.40 | 96.00 |
| led-display | 72.12 | 71.12 | 70.80 | 71.56 | 71.80 | 70.36 | 71.32 | 70.48 | 70.44 | 69.16 | 70.32 | 69.60 |
| lenses | 83.33 | 75.00 | 75.00 | 75.00 | 75.00 | 75.00 | 75.00 | 75.00 | 75.00 | 75.00 | 75.00 | 75.00 |
| letter | 72.61 | 72.49 | 70.06 | 72.38 | 72.27 | 71.95 | 72.25 | 72.24 | 72.22 | 60.57 | 70.26 | 96.53 |
| libras | 64.56 | 61.56 | 61.78 | 62.11 | 65.00 | 62.56 | 62.33 | 62.22 | 62.67 | 61.56 | 61.11 | 78.22 |
| low-res-spect | 89.96 | 88.15 | 88.60 | 88.15 | 88.83 | 88.23 | 87.77 | 87.55 | 88.30 | 88.30 | 88.23 | 90.42 |
| lung-cancer | 62.50 | 61.25 | 57.50 | 62.50 | 62.50 | 62.50 | 62.50 | 62.50 | 57.50 | 60.00 | 58.75 | 56.25 |
| lymphography | 84.59 | 83.24 | 81.62 | 83.78 | 85.14 | 82.16 | 82.97 | 81.89 | 82.43 | 82.16 | 82.43 | 80.27 |
| molec-biol-splice | 84.13 | 82.98 | 82.70 | 82.75 | 82.81 | 82.71 | 82.46 | 82.62 | 82.41 | 81.93 | 81.98 | 85.03 |
| nursery | 90.64 | 89.87 | 89.88 | 89.84 | 90.04 | 89.86 | 89.86 | 89.98 | 89.84 | 89.51 | 89.81 | 99.31 |
| oocytes-merluccius-states-2f | 92.41 | 91.86 | 92.02 | 91.94 | 92.09 | 91.70 | 91.74 | 91.70 | 91.51 | 92.02 | 91.66 | 91.66 |
| oocytes-trisopterus-states-5b | 93.29 | 92.68 | 92.85 | 92.72 | 92.89 | 92.76 | 92.81 | 92.76 | 92.50 | 93.07 | 92.59 | 93.60 |
| optical | 94.98 | 94.78 | 94.64 | 94.82 | 95.05 | 94.92 | 94.79 | 94.75 | 94.74 | 94.31 | 94.76 | 97.28 |
| page-blocks | 96.83 | 96.43 | 96.21 | 96.39 | 96.62 | 96.24 | 96.24 | 96.12 | 96.29 | 95.96 | 96.05 | 96.64 |
| pendigits | 89.86 | 89.68 | 89.53 | 89.38 | 89.85 | 89.52 | 89.54 | 89.66 | 89.85 | 89.47 | 89.67 | 97.66 |
| pittsburg-bridges-MATERIAL | 88.30 | 87.92 | 87.17 | 86.79 | 87.55 | 86.42 | 86.42 | 86.79 | 87.55 | 88.68 | 88.68 | 85.66 |
| pittsburg-bridges-REL-L | 69.80 | 67.45 | 67.06 | 67.45 | 68.63 | 68.24 | 63.92 | 67.84 | 65.49 | 66.27 | 64.31 | 69.80 |
| pittsburg-bridges-SPAN | 79.57 | 76.09 | 69.57 | 74.35 | 80.00 | 77.39 | 74.35 | 77.39 | 75.65 | 74.78 | 75.22 | 70.00 |
| pittsburg-bridges-TYPE | 65.77 | 63.85 | 65.77 | 62.31 | 65.38 | 62.31 | 62.69 | 63.46 | 64.62 | 58.85 | 63.46 | 61.15 |
| plant-margin | 78.67 | 77.70 | 76.12 | 77.90 | 78.53 | 78.03 | 78.07 | 78.00 | 69.50 | 58.50 | 64.80 | 80.12 |
| plant-shape | 55.25 | 54.22 | 49.47 | 54.65 | 54.62 | 53.65 | 53.85 | 53.40 | 50.37 | 40.92 | 47.10 | 67.00 |
| plant-texture | 80.50 | 78.15 | 79.32 | 78.90 | 79.77 | 79.05 | 78.92 | 78.97 | 76.35 | 70.46 | 75.02 | 80.88 |
| post-operative | 67.11 | 66.67 | 63.56 | 63.11 | 66.67 | 64.89 | 62.67 | 64.44 | 54.67 | 57.33 | 55.56 | 63.11 |
| primary-tumor | 49.09 | 46.55 | 43.64 | 45.70 | 47.76 | 47.52 | 45.33 | 46.55 | 45.21 | 41.21 | 42.42 | 45.21 |
| seeds | 94.29 | 93.71 | 93.52 | 94.29 | 94.29 | 93.52 | 93.71 | 93.52 | 92.76 | 90.48 | 92.19 | 91.43 |
| semeion | 92.69 | 92.19 | 91.56 | 91.91 | 92.46 | 91.48 | 91.96 | 91.58 | 89.12 | 86.18 | 85.43 | 94.65 |
| soybean | 86.93 | 84.84 | 85.88 | 85.10 | 86.80 | 85.88 | 84.97 | 85.62 | 85.36 | 88.24 | 86.67 | 87.71 |
| statlog-image | 93.11 | 92.28 | 92.16 | 92.28 | 93.65 | 92.92 | 92.24 | 92.31 | 91.76 | 91.90 | 91.26 | 95.64 |
| statlog-landsat | 83.99 | 82.20 | 81.30 | 81.99 | 81.73 | 81.58 | 81.68 | 81.54 | 81.91 | 80.37 | 81.15 | 91.89 |
| statlog-shuttle | 96.54 | 93.49 | 93.41 | 93.50 | 94.55 | 93.88 | 93.61 | 93.63 | 93.12 | 89.87 | 92.49 | 99.90 |
| statlog-vehicle | 78.68 | 77.68 | 78.53 | 77.78 | 79.10 | 77.73 | 77.73 | 77.92 | 77.78 | 77.07 | 77.68 | 81.89 |
| steel-plates | 71.61 | 71.03 | 69.75 | 70.95 | 71.53 | 70.76 | 70.62 | 70.52 | 70.47 | 70.02 | 70.62 | 75.03 |
| synthetic-control | 97.93 | 97.27 | 93.00 | 97.33 | 97.73 | 94.27 | 96.60 | 96.07 | 91.33 | 87.87 | 89.20 | 99.00 |
| teaching | 49.60 | 48.27 | 46.67 | 48.53 | 50.13 | 48.80 | 48.80 | 48.53 | 48.00 | 40.80 | 46.67 | 52.53 |
| thyroid | 96.04 | 94.75 | 94.84 | 94.89 | 96.06 | 95.06 | 95.04 | 94.99 | 95.06 | 95.23 | 94.46 | 96.66 |
| vertebral-column-3clases | 86.97 | 85.55 | 85.16 | 85.81 | 86.19 | 84.52 | 85.16 | 83.87 | 84.00 | 83.61 | 84.77 | 83.35 |
| wall-following | 72.85 | 69.86 | 69.27 | 69.77 | 72.17 | 69.93 | 69.52 | 69.92 | 69.57 | 72.89 | 66.54 | 90.66 |
| waveform | 87.40 | 86.86 | 87.09 | 86.84 | 87.43 | 87.08 | 86.88 | 87.25 | 86.66 | 86.74 | 86.70 | 86.92 |
| waveform-noise | 86.15 | 85.86 | 85.78 | 85.97 | 86.06 | 85.88 | 86.02 | 85.86 | 85.98 | 85.77 | 85.90 | 85.44 |
| wine | 98.88 | 98.43 | 98.88 | 98.88 | 98.88 | 97.75 | 98.43 | 98.20 | 98.88 | 98.88 | 98.65 | 97.75 |
| wine-quality-red | 58.22 | 57.25 | 56.52 | 56.47 | 58.12 | 56.42 | 56.47 | 56.40 | 56.85 | 56.35 | 56.40 | 61.20 |
| wine-quality-white | 53.87 | 53.13 | 53.07 | 53.32 | 54.02 | 53.74 | 53.51 | 54.02 | 53.56 | 47.10 | 53.27 | 60.21 |
| yeast | 60.75 | 59.95 | 59.65 | 59.62 | 61.11 | 60.49 | 60.19 | 60.38 | 60.16 | 52.26 | 59.97 | 60.75 |
| zoo | 96.00 | 96.00 | 96.00 | 96.00 | 96.00 | 96.00 | 96.00 | 96.00 | 96.00 | 95.60 | 96.00 | 96.00 |
| Mean | 77.30 | 75.57 | 75.19 | 75.57 | 76.68 | 75.53 | 75.36 | 75.55 | 74.75 | 72.94 | 74.18 | 79.96 |

TABLE IV: A comparison of multi-class test classification accuracy of the SIC model (15) to $\pi$-weighted convex focal loss [31], LIBLINEAR [12], and LIBSVM [6]. Although the accuracy of TOP1 is less than kernel-based LIBSVM, it still achieves $77.30 \%$ accuracy, which is $0.62 \%$ better than TOP1-FL. Within linear classifiers, the SIC model performs similarly to the convex focal loss having the external $\pi$-weight parameter, i.e., MaxM achieves $75.57 \%$ vs. MaxM-FL achieves $75.55 \%$.
model for each dataset(TOP1) can lead to better performance than the kernel-based LIBSVM. Specifically, TOP1 achieves a test classification accuracy of $83.96 \%$, which is $0.74 \%$ better than the accuracy of LIBSVM and $0.16 \%$ better than the accuracy of TOP1-FL of the convex focal loss. In multi-class classification problems, although the test accuracy of TOP1 is lower than that of kernel-based LIBSVM, it achieves an accuracy of $77.30 \%$, which is $0.62 \%$ better than the accuracy of TOP1-FL of the convex focal loss. Last but not least, the proposed SIC model, which includes an ( $\alpha_{+}, \alpha_{-}$)-matrix parameter, could be a valuable tool for analyzing various structures of datasets, such as $r_{s c}$-inconsistency and multi-label structures.

## Appendix

## A. Proof of Theorem III. 2

Let $\alpha=1$, then we have $s_{\alpha, c}(x)=\frac{1}{1+\exp (-x)} \in C^{\infty}(\mathbb{R})$. Therefore, we only consider the case $\alpha \neq 1$. Note that $\nabla^{n} s_{\alpha, c}(x)=0$, for all $x \in \mathbb{R} \backslash \operatorname{dom}\left(\sigma_{\alpha, c}\right)$ and $n=1,2,3, \cdots$. Also, for $x \in \operatorname{int}\left(\operatorname{dom}\left(\sigma_{\alpha, c}\right)\right)$, let $y=1+\frac{x}{c_{\alpha}}>0$ and $a=\frac{1}{1-\alpha}$, then we get $s_{\alpha, c}\left(c_{\alpha}(y-1)\right)=\frac{1}{1+y^{\alpha}}$. Thus, it is not difficult to see $s_{\alpha, c}(x)=$ $\sigma_{\alpha, c}(x) \in C^{\infty}\left(\operatorname{int}\left(\operatorname{dom}\left(\sigma_{\alpha, c}\right)\right)\right)$.

For the continuity of $\nabla^{n} s_{\alpha, c}(x)$ on $x \in \mathbb{R}$, we only need to check $s_{\alpha, c}\left(-c_{\alpha}\right)=0$. Let us assume that (7) is true. (A) When $0 \leq \alpha<1$, $\exp _{\alpha, c}(-x)=0$ at $x=-c_{\alpha}$. Thus, we only need to consider numerator of $F_{n, k}(x)$, i.e., $\left(\exp _{\alpha, c}(-x)\right)^{k-n(1-\alpha)} . F_{n, k}\left(c_{\alpha}\right)=0$, if $k-n(1-\alpha)>0$ for all $k=1,2, \cdots, n$. Now, we get $1>\alpha>1-1 / n$. (B) When $\alpha>1, \exp _{\alpha, c}(-x)=+\infty$ at $x=-c_{\alpha}$. Thus, we have $F_{n, k}\left(-c_{\alpha}\right)=$ $\lim _{x \rightarrow-c_{\alpha}} c A_{n, k}\left(\frac{1}{1-\alpha}\right)\left(\exp _{\alpha, c}(-x)\right)^{-n(1-\alpha)-1}=0$ if $-n(1-\alpha)-1<0$. It means $1<\alpha<1+1 / n$. From (A) and (B), we get $\nabla^{n} s_{\alpha, c}(x) \in C^{n}(\mathbb{R})$ for $\alpha \in\left(1-\frac{1}{n}, 1+\frac{1}{n}\right)$.

Now, we want to show (7) by induction for $x \in \operatorname{int}\left(\operatorname{dom}\left(\sigma_{\alpha, c}\right)\right)$. Let $y=1+\frac{x}{c_{\alpha}}$ and $a=\frac{1}{1-\alpha}$, then (7) becomes

$$
\begin{equation*}
\frac{d^{n}}{d y^{n}} \frac{1}{1+y^{a}}=\sum_{k=1}^{n} B_{n, k}(a) \frac{y^{k a-n}}{\left(1+y^{a}\right)^{k+1}} \tag{36}
\end{equation*}
$$

where $B_{n, k}(a)=(-1)^{n+k} k!\sum_{l=0}^{n}\left[\begin{array}{l}n \\ l\end{array}\right]\left\{\begin{array}{l}l \\ k\end{array}\right\}(-a)^{l}$.
(I) Let $n=1$. Then the left-hand side of (36) is $\frac{d}{d y} \frac{1}{1+y^{a}}=\frac{-a y^{a-1}}{\left(1+y^{a}\right)^{2}}$. The right-hand side is $B_{1,1}(a) \frac{x^{a-1}}{\left(1+x^{a}\right)^{2}}$ where $B_{1,1}(a)=\left[\begin{array}{l}1 \\ 0\end{array}\right]\left\{\begin{array}{l}0 \\ 1\end{array}\right\}+\left[\begin{array}{l}1 \\ 1\end{array}\right]\left\{\begin{array}{l}1 \\ 1\end{array}\right\}(-a)=-a$. For the computation of the Stirling number of the first and second kind, we use the following convention and rule in [17]: $\left\{\begin{array}{l}0 \\ 0\end{array}\right\}=\left[\begin{array}{l}0 \\ 0\end{array}\right]=1$ and $\left\{\begin{array}{l}a \\ 0\end{array}\right\}=\left[\begin{array}{l}a \\ 0\end{array}\right]=0$ for $a \geq 1$. Also, we have $\left\{\begin{array}{l}a \\ 1\end{array}\right\}=1$ and $\left[\begin{array}{l}a \\ 1\end{array}\right]=(a-1)$ ! with $0!=1$, for $a \geq 1$. Additionally, $\left\{\begin{array}{l}a \\ b\end{array}\right\}=\left[\begin{array}{l}a \\ b\end{array}\right]=0$ if $b>a \geq 0$.
(II) For $n>1$, let (36) be true. Then, for $n+1$, we need to show that

$$
\begin{equation*}
\frac{d}{d y}\left(\sum_{k=1}^{n} B_{n, k}(a) \frac{y^{a k-n}}{\left(1+y^{a}\right)^{k+1}}\right)=\sum_{k=1}^{n+1} B_{n+1, k}(a) \frac{y^{a k-(n+1)}}{\left(1+y^{a}\right)^{k+1}} \tag{37}
\end{equation*}
$$

where

$$
\begin{equation*}
\frac{d}{d y}\left(\sum_{k=1}^{n} B_{n, k}(a) \frac{y^{a k-n}}{\left(1+y^{a}\right)^{k+1}}\right)=\sum_{k=1}^{n} B_{n, k}(a)\left(\frac{(a k-n) y^{a k-(n+1)}}{\left(1+y^{a}\right)^{k+1}}-(k+1) \frac{a y^{a(k+1)-(n+1)}}{\left(1+y^{a}\right)^{k+2}}\right) . \tag{38}
\end{equation*}
$$

From (37) and (38), we get

$$
\begin{equation*}
B_{n+1, k}(a)=(-k a) B_{n, k-1}(a)+(k a-n) B_{n, k}(a) \tag{39}
\end{equation*}
$$

where $B_{n, 0}(a)=B_{n, n+1}(a)=0$. It comes from the rule of the Stirling number in $(\mathbf{I})$. Now, we only need to prove (39). The left-hand side of (39) is

$$
\begin{aligned}
B_{n+1, k}(a) & =(-1)^{k} k!\sum_{l=0}^{n+1}\left[\begin{array}{c}
n+1 \\
l
\end{array}\right]\left\{\begin{array}{l}
l \\
k
\end{array}\right\}(-1)^{n+1-l} a^{l} \\
& \left.=(-1)^{k} k!\left(\left\{\begin{array}{c}
n+1 \\
k
\end{array}\right\} a^{n+1}+\sum_{l=1}^{n}\left(\begin{array}{l}
n \\
n \\
l
\end{array}\right]+\left[\begin{array}{c}
n \\
l-1
\end{array}\right]\right)\left\{\begin{array}{l}
l \\
k
\end{array}\right\}(-1)^{n+1-l} a^{l}\right)
\end{aligned}
$$

where $\left[\begin{array}{c}n+1 \\ 0\end{array}\right]=0,\left[\begin{array}{c}n+1 \\ n+1\end{array}\right]=1$, and $\left[\begin{array}{c}n+1 \\ l\end{array}\right]=n\left[\begin{array}{c}n \\ l\end{array}\right]+\left[\begin{array}{c}n \\ l-1\end{array}\right]$ (see the rule of the Stirling number in (I) and [17]). By using the equivalence $\left\{\begin{array}{c}l+1 \\ k\end{array}\right\}=\left\{\begin{array}{c}l \\ k-1\end{array}\right\}+k\left\{\begin{array}{l}l \\ k\end{array}\right\}$ and $\left[\begin{array}{c}n \\ 0\end{array}\right]=0$, the right-hand side of (39) is simplified to the following equation.

$$
(-k a) B_{n, k-1}(a)+(k a-n) B_{n, k}(a)=(-1)^{k} k!\sum_{l=0}^{n}\left(a\left\{\begin{array}{c}
l+1 \\
k
\end{array}\right\}-n\left\{\begin{array}{c}
l \\
k
\end{array}\right\}\right)\left[\begin{array}{c}
n \\
l
\end{array}\right](-1)^{n-l} a^{l}
$$

By dividing $(-1)^{k} k$ ! and adding $\sum_{l=1}^{n} n\left[\begin{array}{c}n \\ l\end{array}\right]\left\{\begin{array}{l}l \\ k\end{array}\right\}(-1)^{n-l} a^{l}$ on both sides, we obtain the equivalence (39).

## B. The structure of the dataset and classification results of all-class

We summarize the structure of 118 datasets used in our experiments and present classification accuracy of the SIC model and $\pi$-weighted convex focal loss of all-class. Table $V$ summarizes two-class datasets. For each dataset, we describe the number of instances, the size of training dataset, the size of test dataset, the size of class, and the feature dimension. Additionally, we show imbalancedness, i.e., $r_{c} / r_{c} \mathbf{T} / r_{c} \mathbf{T e}$ for class-imbalance ratio of combined/training/test dataset and $r_{s c} / r_{s c} \mathbf{T} / r_{s c} \mathbf{T e}$ for scale-class-imbalance ratio of combined/training/test dataset. Table VI summarizes multi-class datasets. For each dataset, we describe the number of instances, the size of training dataset, the size of test dataset, the feature dimension, and the size of class. Additionally, we show imbalancedness, i.e., minimum and maximum of $r_{c}$ for combined/training dataset: $r_{c} \mathbf{m} / r_{c} \mathbf{M} / r_{c} \mathbf{T m} / r_{c} \mathbf{T M}$. Also, minimum and maximum of $r_{s c}$ for combined/training dataset: $r_{s c} \mathbf{m} / r_{s c} \mathbf{M} / r_{s c} \mathbf{T m} / r_{s c} \mathbf{T M}$.

Figure 9 presents classification accuracy matrix and the corresponding histogram of all-class for $20 \times 20$ SIC models and $19 \times 8$ convex focal losses. The test classification accuracy of all SIC models ranges between $76.88 \%$ and $78.56 \%$. On the other hand, the test classification accuracy of all convex focal losses ranges between $71.04 \%$ and $78.39 \%$.

## Acknowledgments

H. Woo is supported by Logitron X.

## REFERENCES

[1] S. Amari, Information geometry and its applications, Springer, 2016.
[2] F. R. BACH, D. HECKERMAN, AND E. HORVITZ, "Considering cost asymmetry in learning classifiers,", Journal of Machine Learning Research, vol. 7, pp. 1713-1741, 2006.
[3] A. DE BRÉBISSON AND P. VinCENT, "An exploration of softmax alternatives belonging to the spherical loss family", arXiv:1511.05042v3, 2016.
[4] J. BYRD AND Z. C. Lipton, "What is the effect of importance weighting in deep learning", Proceedings of the 36th International Conference on Machine Learning, 2019.
[5] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss", 33rd Conference on Neural Information Processing Systems (NeurIPS 2019), 2019.
[6] C.-C. Chang and C.-J. Lin, "LIBSVM : a library for support vector machines", ACM Transactions on Intelligent Systems and Technology, vol. 2, pp. 27:1-27:27, 2011.
[7] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, S. Belongie, "Class-balanced loss based on effective number of samples", arXiv:1901.555551, 28 September 2018.
[8] M. F.-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?", Journal of Machine Learning Research, vol. 15, pp. 3133-3181, 2014.
[9] N. Ding and S.V.N. Vishwanathan, "t-logistic regression", Advances in Neural Information Processing Systems 23, 2010.
[10] S. R. Dubey, S. K. Singh, and B. B. Chaudhuri, "Activation functions in deep learning: A comprehensive survey and benchmark", arXiv:2109.14545v3, 2022.
[11] G. F. Elsayed, D. Krishnan, H. Mobahi, K. Regan, and S. Bengio, "Large margin deep networks for classification", 32nd Conference on Neural Information Processing Systems (NeurIPS), 2018.
[12] R.-E. FAN, K.-W. Chang, C.-J. Hsieh, X.-R. WANG, and C.-J. Lin, "LIBLINEAR: A library for large linear classification", Journal of Machine Learning Research, vol. 9, pp. 1871-1874, 2008.
[13] C. FANG, H. He, Q. Long, And W. J. Su, "Exploring deep neural networks via layer-peeled model: minority collapse in imbalanced training", Proc. Natl. Acad. Sci. U.S.A., vol. 118, pp. 1-12, 2021.
[14] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, Learning from imbalanced datasets, Springer-Verlag, 2018.
[15] L. Galli and C.-J. Lin, "A study on truncated Newton methods for linear classification", IEEE Transactions on Neural Networks and Learning Systems, vol. 33, pp. 2828-2841, 2022.

|  | instance | train | test | dim | class | $r_{c}$ | $r_{s c}$ | $r_{c} \mathbf{T}$ | $r_{s c} \mathbf{T}$ | $r_{c} \mathbf{T e}$ | $r_{s c} \mathbf{T e}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| acute-inflammation | 120 | 60 | 60 | 6 | 2 | 1.03 | 1.01 | 1 | 0.95 | 1.07 | 1.06 |
| acute-nephritis | 120 | 60 | 60 | 6 | 2 | 1.40 | 1.14 | 1.40 | 1.14 | 1.40 | 1.14 |
| adult | 48842 | 32561 | 16281 | 14 | 2 | 3.18 | 2.10 | 3.15 | 2.09 | 3.23 | 2.14 |
| balloons | 16 | 8 | 8 | 4 | 2 | 1.29 | 1.18 | 1 | 0.72 | 1.67 | 2.15 |
| bank | 4521 | 2261 | 2260 | 16 | 2 | 7.68 | 4.43 | 7.66 | 4.38 | 7.69 | 4.45 |
| blood | 748 | 374 | 374 | 4 | 2 | 3.20 | 2.63 | 3.20 | 2.41 | 3.20 | 2.80 |
| breast-cancer | 286 | 143 | 143 | 9 | 2 | 2.36 | 1.92 | 2.33 | 1.91 | 2.40 | 1.89 |
| breast-cancer-wisc | 699 | 350 | 349 | 9 | 2 | 1.90 | 1.12 | 1.89 | 1.13 | 1.91 | 1.11 |
| breast-cancer-wisc-diag | 569 | 285 | 284 | 30 | 2 | 1.68 | 1.06 | 1.69 | 0.99 | 1.68 | 1.14 |
| breast-cancer-wisc-prog | 198 | 99 | 99 | 33 | 2 | 3.21 | 2.06 | 3.12 | 2.42 | 3.30 | 1.98 |
| chess-krvkp | 3196 | 1598 | 1598 | 36 | 2 | 0.91 | 0.95 | 0.92 | 0.97 | 0.91 | 0.95 |
| congressional-voting | 435 | 218 | 217 | 16 | 2 | 1.59 | 1.53 | 1.60 | 1.54 | 1.58 | 1.51 |
| conn-bench-sonar-mines-rocks | 208 | 104 | 104 | 60 | 2 | 1.14 | 1.04 | 1.12 | 1.02 | 1.17 | 1.06 |
| connect-4 | 67557 | 33779 | 33778 | 42 | 2 | 3.06 | 2.94 | 3.06 | 2.95 | 3.06 | 2.92 |
| credit-approval | 690 | 345 | 345 | 15 | 2 | 0.80 | 0.90 | 0.81 | 0.99 | 0.80 | 0.83 |
| cylinder-bands | 512 | 256 | 256 | 35 | 2 | 0.64 | 0.75 | 0.64 | 0.79 | 0.64 | 0.74 |
| echocardiogram | 131 | 66 | 65 | 10 | 2 | 2.05 | 1.47 | 2 | 1.52 | 2.10 | 1.44 |
| fertility | 100 | 50 | 50 | 9 | 2 | 7.33 | 5.47 | 7.33 | 4.96 | 7.33 | 6.07 |
| haberman-survival | 306 | 153 | 153 | 3 | 2 | 2.78 | 2.53 | 2.73 | 2.22 | 2.83 | 2.79 |
| heart-hungarian | 294 | 147 | 147 | 12 | 2 | 1.77 | 1.26 | 1.77 | 1.37 | 1.77 | 1.17 |
| hepatitis | 155 | 78 | 77 | 19 | 2 | 0.26 | 0.52 | 0.26 | 0.65 | 0.26 | 0.44 |
| hill-valley | 606 | 303 | 303 | 100 | 2 | 1.03 | 1.03 | 1.02 | 0.84 | 1.03 | 0.83 |
| horse-colic | 368 | 300 | 68 | 25 | 2 | 1.71 | 1.30 | 1.75 | 1.31 | 1.52 | 1.22 |
| ilpd-indian-liver | 583 | 292 | 291 | 9 | 2 | 2.49 | 2.04 | 2.48 | 2.07 | 2.51 | 2.06 |
| ionosphere | 351 | 176 | 175 | 33 | 2 | 0.56 | 0.81 | 0.56 | 0.86 | 0.56 | 0.83 |
| magic | 19020 | 9510 | 9510 | 10 | 2 | 1.84 | 1.51 | 1.84 | 1.51 | 1.84 | 1.52 |
| miniboone | 130064 | 65032 | 65032 | 50 | 2 | 0.39 | 0.55 | 0.39 | 0.55 | 0.39 | 0.55 |
| molec-biol-promoter | 106 | 53 | 53 | 57 | 2 | 1 | 1 | 0.96 | 1.01 | 1.04 | 0.99 |
| mammographic | 961 | 481 | 480 | 5 | 2 | 1.16 | 1.09 | 1.16 | 1.11 | 1.16 | 1.06 |
| mushroom | 8124 | 4062 | 4062 | 21 | 2 | 1.07 | 1.03 | 1.07 | 1.05 | 1.07 | 1 |
| musk-1 | 476 | 238 | 238 | 166 | 2 | 1.30 | 1.07 | 1.29 | 1.53 | 1.31 | 0.93 |
| musk-2 | 6598 | 3299 | 3299 | 166 | 2 | 5.49 | 1.74 | 5.48 | 1.78 | 5.49 | 1.73 |
| oocytes-merluccius-nucleus-4d | 1022 | 511 | 511 | 41 | 2 | 0.49 | 0.59 | 0.49 | 0.55 | 0.49 | 0.62 |
| oocytes-trisopterus-nucleus-2f | 912 | 456 | 456 | 25 | 2 | 0.73 | 0.79 | 0.73 | 0.85 | 0.73 | 0.71 |
| ozone | 2536 | 1268 | 1268 | 72 | 2 | 33.74 | 6.98 | 33.27 | 7 | 34.22 | 7.31 |
| parkinsons | 195 | 98 | 97 | 22 | 2 | 0.33 | 0.73 | 0.32 | 0.92 | 0.33 | 0.56 |
| pima | 768 | 384 | 384 | 8 | 2 | 1.87 | 1.53 | 1.87 | 1.51 | 1.87 | 1.53 |
| pittsburg-bridges-T-OR-D | 102 | 51 | 51 | 7 | 2 | 6.29 | 4.10 | 6.29 | 4.71 | 6.29 | 3.41 |
| planning | 182 | 91 | 91 | 12 | 2 | 2.50 | 2.46 | 2.50 | 2.39 | 2.50 | 2.45 |
| ringnorm | 7400 | 3700 | 3700 | 20 | 2 | 0.98 | 0.99 | 0.98 | 1 | 0.98 | 0.98 |
| spambase | 4601 | 2301 | 2300 | 57 | 2 | 1.54 | 1.16 | 1.54 | 1.16 | 1.54 | 1.17 |
| spect | 265 | 79 | 186 | 22 | 2 | 1.41 | 1.21 | 2.04 | 1.41 | 1.21 | 1.11 |
| spectf | 267 | 80 | 187 | 44 | 2 | 0.26 | 0.52 | 1 | 1 | 0.09 | 0.26 |
| statlog-australian-credit | 690 | 345 | 345 | 14 | 2 | 0.47 | 0.49 | 0.47 | 0.49 | 0.47 | 0.48 |
| statlog-german-credit | 1000 | 500 | 500 | 24 | 2 | 2.33 | 1.82 | 2.33 | 1.81 | 2.33 | 1.84 |
| statlog-heart | 270 | 135 | 135 | 13 | 2 | 1.25 | 1.09 | 1.25 | 1.15 | 1.25 | 1.02 |
| tic-tac-toe | 958 | 479 | 479 | 9 | 2 | 0.53 | 0.60 | 0.53 | 0.61 | 0.53 | 0.59 |
| titanic | 2201 | 1101 | 1100 | 3 | 2 | 2.10 | 1.76 | 2.09 | 1.75 | 2.10 | 1.77 |
| trains | 10 | 5 | 5 | 29 | 2 | 1 | 1 | 0.67 | 0.70 | 1.50 | 1.21 |
| twonorm | 7400 | 3700 | 3700 | 20 | 2 | 1 | 1 | 1 | 1 | 1 | 1 |
| vertebral-column-2clases | 310 | 155 | 155 | 6 | 2 | 2.10 | 1.56 | 2.10 | 1.51 | 2.10 | 1.63 |

TABLE V: Two-class datasets in [8], [55]. Note that $r_{c}$ is the class-imbalance ratio and $r_{s c}$ is the scale-classimbalance ratio defined in (1). Additionally, $r_{c} \mathbf{T}$ and $r_{s c} \mathbf{T}$ (or $r_{c} \mathbf{T e}$ and $r_{s c} \mathbf{T e}$ ) are $r_{c}$ and $r_{s c}$ of training dataset (or test dataset).
[16] S. Garcia, J. Luengo, and F. Herrera, Data preprocessing in data mining. Springer-Verlag, 2015.
[17] R. Graham, D. Knuth, and O. Patashnik, Concrete Mathematics: A foundation for computer science, Second Edition, Addison-Wesley, 1994.
[18] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks", Proceedings of the 34th International Conference on Machine Learning, 2017.
[19] W. W. HAGER AND H. ZHANG, "A new conjugate gradient method with guaranteed descent and an efficient line search", SIAM Journal on Optimization, vol. 16, pp. 170-192, 2005.
[20] W. W. Hager and H. Zhang, "Algorithm 851: CG_DESCENT, a conjugate gradient method with guaranteed descent", ACM Transactions on Mathematical Software, vol. 32, pp. 113-137, 2006.
[21] H. He and E. A. Garcia, "Learning from imbalanced data", IEEE Transactions on Knowledge and Data Engineering, vol. 21, pp. 1263-1284, 2009.
[22] J.-B. Hiriart-Urruty and C. Lemarechal, Convex Analysis and Minimization Algorithms I. Springer-Verlag, 1996.
[23] S. IOFFE AND C. SZEGEDY, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", arXiv:1502.03167v3, 2015.
[24] K. JANOCHA AND W. M. CZARNECKI, "On loss functions for deep neural networks in classification", arXiv:1702.05659v1, 2017.
[25] Z. Ji, M. DUdik, R. E. Schapire, and M. TElGarsky, "Gradient descent follows the regularization path for general losses", $33 r d$ Annual Conference on Learning Theory, 2020.
[26] Y. Jiang, D. Krishnan, H. Mobahi, S. Bengio, "Predicting the generalization gap in deep networks with margin distributions", arXiv:1810.00113v1, 28 September 2018.
[27] J. M. Johnson and T. M. Khoshgoftana, "Survey on deep learning with class imbalance", Journal of Big Data, vol. 6, pp. 1-54., 2019.
[28] B. Jorgensen, The Theory of Dispersion Models, Chapman \& Hall, 1997.

|  | inst. | train | test | dim | cls | $r_{C} \mathbf{m}$ | $r_{c} \mathbf{M}$ | $r_{s c} \mathbf{m}$ | $r_{s c} \mathbf{M}$ | $r_{c} \mathbf{T m}$ | $r_{c} \mathbf{T M}$ | $r_{s c} \mathbf{T m}$ | $r_{s c} \mathbf{T M}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| abalone | 4177 | 2089 | 2088 | 8 | 3 | 0.46 | 0.53 | 0.53 | 0.82 | 0.46 | 0.53 | 0.54 | 0.82 |
| annealing | 798 | 399 | 399 | 31 | 5 | 0.01 | 3.20 | 0.06 | 2.06 | 0.01 | 3.20 | 0.11 | 2.12 |
| arrhythmia | 452 | 226 | 226 | 262 | 13 | 0 | 1.18 | 0.05 | 1.04 | 0 | 1.11 | 0.06 | 1.02 |
| audiology-std | 196 | 171 | 25 | 59 | 18 | 0.01 | 0.32 | 0.05 | 0.59 | 0.01 | 0.36 | 0.05 | 0.59 |
| balance-scale | 625 | 313 | 312 | 4 | 3 | 0.09 | 0.85 | 0.09 | 0.91 | 0.09 | 0.85 | 0.09 | 0.94 |
| breast-tissue | 106 | 53 | 53 | 9 | 6 | 0.15 | 0.26 | 0.31 | 0.71 | 0.15 | 0.26 | 0.27 | 0.58 |
| car | 1728 | 864 | 864 | 6 | 4 | 0.04 | 2.34 | 0.08 | 1.77 | 0.04 | 2.32 | 0.08 | 1.79 |
| cardiotocography-10clases | 2126 | 1063 | 1063 | 21 | 10 | 0.03 | 0.37 | 0.07 | 0.55 | 0.03 | 0.37 | 0.07 | 0.54 |
| cardiotocography-3clases | 2126 | 1063 | 1063 | 21 | 3 | 0.09 | 3.51 | 0.34 | 1.85 | 0.09 | 3.50 | 0.31 | 1.89 |
| chess-krvk | 28056 | 14028 | 14028 | 6 | 18 | 0 | 0.19 | 0 | 0.24 | 0 | 0.19 | 0 | 0.24 |
| conn-bench-vowel-deterding | 528 | 264 | 264 | 11 | 11 | 0.10 | 0.10 | 0.11 | 0.25 | 0.10 | 0.10 | 0.12 | 0.26 |
| contrac | 1473 | 737 | 736 | 9 | 3 | 0.29 | 0.75 | 0.37 | 0.78 | 0.29 | 0.74 | 0.39 | 0.77 |
| dermatology | 366 | 183 | 183 | 34 | 6 | 0.06 | 0.44 | 0.39 | 0.90 | 0.06 | 0.43 | 0.40 | 0.87 |
| ecoli | 336 | 168 | 168 | 7 | 8 | 0.01 | 0.74 | 0.01 | 0.89 | 0.01 | 0.71 | 0.01 | 0.84 |
| energy-y1 | 768 | 384 | 384 | 8 | 3 | 0.22 | 0.88 | 0.31 | 0.97 | 0.22 | 0.87 | 0.29 | 0.99 |
| energy-y2 | 768 | 384 | 384 | 8 | 3 | 0.33 | 0.99 | 0.58 | 1 | 0.33 | 0.99 | 0.60 | 0.95 |
| flags | 194 | 97 | 97 | 28 | 8 | 0.02 | 0.45 | 0.06 | 0.68 | 0.02 | 0.43 | 0.05 | 0.66 |
| glass | 214 | 107 | 107 | 9 | 6 | 0.04 | 0.55 | 0.11 | 0.60 | 0.05 | 0.51 | 0.10 | 0.57 |
| hayes-roth | 160 | 132 | 28 | 3 | 3 | 0.24 | 0.68 | 0.40 | 0.74 | 0.29 | 0.63 | 0.46 | 0.69 |
| heart-cleveland | 303 | 152 | 151 | 13 | 5 | 0.04 | 1.18 | 0.11 | 1.07 | 0.05 | 1.14 | 0.14 | 1.14 |
| heart-switzerland | 123 | 62 | 61 | 12 | 5 | 0.04 | 0.64 | 0.06 | 0.67 | 0.05 | 0.63 | 0.08 | 0.66 |
| heart-va | 200 | 100 | 100 | 12 | 5 | 0.05 | 0.39 | 0.10 | 0.46 | 0.05 | 0.37 | 0.09 | 0.48 |
| image-segmentation | 2310 | 210 | 2100 | 18 | 7 | 0.17 | 0.17 | 0.27 | 0.69 | 0.17 | 0.17 | 0.27 | 0.69 |
| iris | 150 | 75 | 75 | 4 | 3 | 0.50 | 0.50 | 0.58 | 0.82 | 0.50 | 0.50 | 0.56 | 0.84 |
| led-display | 1000 | 500 | 500 | 7 | 10 | 0.09 | 0.12 | 0.18 | 0.30 | 0.09 | 0.13 | 0.19 | 0.29 |
| lenses | 24 | 12 | 12 | 4 | 3 | 0.20 | 1.67 | 0.35 | 1.37 | 0.20 | 1.40 | 0.35 | 1.39 |
| letter | 20000 | 10000 | 10000 | 16 | 26 | 0.04 | 0.04 | 0.06 | 0.14 | 0.04 | 0.04 | 0.06 | 0.14 |
| libras | 360 | 180 | 180 | 90 | 15 | 0.07 | 0.07 | 0.12 | 0.54 | 0.07 | 0.07 | 0.13 | 0.51 |
| low-res-spect | 531 | 266 | 265 | 100 | 9 | 0 | 1.08 | 0.05 | 1.01 | 0 | 1.06 | 0.04 | 0.98 |
| lung-cancer | 32 | 16 | 16 | 56 | 3 | 0.39 | 0.68 | 0.78 | 0.87 | 0.45 | 0.60 | 0.72 | 0.87 |
| lymphography | 148 | 74 | 74 | 18 | 4 | 0.01 | 1.21 | 0.08 | 1.10 | 0.01 | 1.18 | 0.08 | 1.04 |
| molec-biol-splice | 3190 | 1595 | 1595 | 60 | 3 | 0.32 | 1.08 | 0.51 | 1.05 | 0.32 | 1.08 | 0.51 | 1.07 |
| nursery | 12960 | 6480 | 6480 | 8 | 5 | 0 | 0.50 | 0 | 0.67 | 0 | 0.50 | 0 | 0.67 |
| oocytes-merluccius-states-2f | 1022 | 511 | 511 | 25 | 3 | 0.06 | 2.19 | 0.37 | 1.14 | 0.06 | 2.17 | 0.37 | 1.07 |
| oocytes-trisopterus-states-5b | 912 | 456 | 456 | 32 | 3 | 0.02 | 1.36 | 0.08 | 1.04 | 0.02 | 1.35 | 0.09 | 1.08 |
| optical | 5620 | 3823 | 1797 | 62 | 10 | 0.11 | 0.11 | 0.30 | 0.51 | 0.11 | 0.11 | 0.31 | 0.53 |
| page-blocks | 5473 | 2737 | 2736 | 10 | 5 | 0.01 | 8.77 | 0.06 | 4.07 | 0.01 | 8.74 | 0.06 | 4.01 |
| pendigits | 10992 | 7494 | 3498 | 16 | 10 | 0.11 | 0.12 | 0.23 | 0.44 | 0.11 | 0.12 | 0.25 | 0.44 |
| pittsburg-bridges-MATERIAL | 106 | 53 | 53 | 7 | 3 | 0.12 | 2.93 | 0.20 | 1.67 | 0.13 | 2.79 | 0.29 | 1.37 |
| pittsburg-bridges-REL-L | 103 | 52 | 51 | 7 | 3 | 0.17 | 1.06 | 0.24 | 1.04 | 0.18 | 1 | 0.25 | 1.20 |
| pittsburg-bridges-SPAN | 92 | 46 | 46 | 7 | 3 | 0.31 | 1.09 | 0.45 | 1.07 | 0.31 | 1.09 | 0.39 | 1.08 |
| pittsburg-bridges-TYPE | 105 | 53 | 52 | 7 | 6 | 0.11 | 0.72 | 0.14 | 0.78 | 0.10 | 0.66 | 0.17 | 0.79 |
| plant-margin | 1600 | 800 | 800 | 64 | 100 | 0.01 | 0.01 | 0.03 | 0.14 | 0.01 | 0.01 | 0.03 | 0.14 |
| plant-shape | 1600 | 800 | 800 | 64 | 100 | 0.01 | 0.01 | 0.01 | 0.25 | 0.01 | 0.01 | 0.01 | 0.25 |
| plant-texture | 1599 | 800 | 799 | 64 | 100 | 0.01 | 0.01 | 0.03 | 0.11 | 0.01 | 0.01 | 0.03 | 0.12 |
| post-operative | 90 | 45 | 45 | 8 | 3 | 0.02 | 2.46 | 0.06 | 2.31 | 0.02 | 2.46 | 0.06 | 1.95 |
| primary-tumor | 330 | 165 | 165 | 17 | 15 | 0.02 | 0.34 | 0.04 | 0.55 | 0.02 | 0.32 | 0.04 | 0.50 |
| seeds | 210 | 105 | 105 | 7 | 3 | 0.50 | 0.50 | 0.65 | 0.86 | 0.50 | 0.50 | 0.69 | 0.87 |
| semeion | 1593 | 797 | 796 | 256 | 10 | 0.11 | 0.11 | 0.52 | 0.74 | 0.11 | 0.11 | 0.52 | 0.69 |
| soybean | 307 | 154 | 153 | 35 | 18 | 0.01 | 0.15 | 0.10 | 0.49 | 0.01 | 0.15 | 0.10 | 0.49 |
| statlog-image | 2310 | 1155 | 1155 | 18 | 7 | 0.17 | 0.17 | 0.27 | 0.69 | 0.17 | 0.17 | 0.27 | 0.68 |
| statlog-landsat | 6435 | 4435 | 2000 | 36 | 6 | 0.11 | 0.31 | 0.26 | 0.83 | 0.10 | 0.32 | 0.25 | 0.84 |
| statlog-shuttle | 58000 | 43500 | 14500 | 9 | 7 | 0 | 3.67 | 0 | 1.67 | 0 | 3.63 | 0 | 1.67 |
| statlog-vehicle | 846 | 423 | 423 | 18 | 4 | 0.31 | 0.35 | 0.48 | 0.62 | 0.31 | 0.35 | 0.35 | 0.72 |
| steel-plates | 1941 | 971 | 970 | 27 | 7 | 0.03 | 0.53 | 0.09 | 0.76 | 0.03 | 0.53 | 0.10 | 0.71 |
| synthetic-control | 600 | 300 | 300 | 60 | 6 | 0.20 | 0.20 | 0.28 | 0.85 | 0.20 | 0.20 | 0.29 | 0.87 |
| teaching | 151 | 76 | 75 | 5 | 3 | 0.48 | 0.53 | 0.53 | 0.58 | 0.49 | 0.52 | 0.54 | 0.54 |
| thyroid | 7200 | 3772 | 3428 | 21 | 3 | 0.02 | 12.48 | 0.08 | 5.19 | 0.03 | 12.28 | 0.08 | 4.96 |
| vertebral-column-3clases | 310 | 155 | 155 | 6 | 3 | 0.24 | 0.94 | 0.43 | 0.98 | 0.24 | 0.94 | 0.43 | 0.92 |
| wall-following | 5456 | 2728 | 2728 | 24 | 4 | 0.06 | 0.68 | 0.24 | 0.78 | 0.06 | 0.68 | 0.23 | 0.80 |
| waveform | 5000 | 2500 | 2500 | 21 | 3 | 0.49 | 0.51 | 0.75 | 0.85 | 0.49 | 0.51 | 0.74 | 0.85 |
| waveform-noise | 5000 | 2500 | 2500 | 40 | 3 | 0.49 | 0.51 | 0.76 | 0.84 | 0.49 | 0.51 | 0.77 | 0.86 |
| wine | 178 | 89 | 89 | 13 | 3 | 0.37 | 0.66 | 0.78 | 0.87 | 0.37 | 0.65 | 0.76 | 0.89 |
| wine-quality-red | 1599 | 800 | 799 | 11 | 6 | 0.01 | 0.74 | 0.02 | 0.81 | 0.01 | 0.74 | 0.02 | 0.81 |
| wine-quality-white | 4898 | 2449 | 2449 | 11 | 7 | 0 | 0.81 | 0 | 0.82 | 0 | 0.81 | 0 | 0.82 |
| yeast | 1484 | 742 | 742 | 8 | 10 | 0 | 0.45 | 0.03 | 0.51 | 0 | 0.45 | 0.03 | 0.51 |
| zoo | 101 | 51 | 50 | 16 | 7 | 0.04 | 0.68 | 0.13 | 0.93 | 0.04 | 0.65 | 0.13 | 0.91 |

TABLE VI: Multi-class datasets in [8], [55]. Here, 0 means $<0.01$. Note that $r_{c} \mathbf{m} / \mathbf{M}$ is the minimum/maximum class-imbalance ratio of datasets and $r_{s c} \mathbf{m} / \mathbf{M}$ is the minimum/maximum scale-class-imbalance ratio defined in (1). Additionally, $r_{c} \mathbf{T m} / \mathbf{M}$ and $r_{s c} \mathbf{T m} / \mathbf{M}$ are the corresponding imbalance ratios for training datasets. OVA strategy is used for $r_{c}$ and $r_{s c}$.
$2 / 3$ 77.96 78.11 78.3178 .3178 .3978 .4678 .4178 .3678 .3278 .2578 .4578 .4478 .4278 .3878 .3678 .2578 .2678 .0777 .7577 .26
$-\quad 3 / 477.8578 .0678 .2778 .3278 .3178 .3978 .3778 .3878 .4878 .4178 .4278 .4578 .2978 .2778 .2978 .2678 .1778 .0177 .7577 .24 \mid$
$8 / 977.8977 .9778 .0678 .0578 .2378 .3178 .378 .3378 .3478 .3778 .2978 .2378 .3278 .2578 .1978 .1277 .9577 .8277 .5977 .12$
$9 / 878.0678 .1378 .2278 .378 .4278 .4478 .4278 .3678 .3478 .4178 .4378 .478 .3878 .3978 .2578 .2578 .2277 .9877 .7677 .26 \mid$
$8 / 778.1278 .27$ 78.3 78.3378 .3778 .4378 .4578 .5678 .3378 .3578 .4578 .4178 .4478 .4378 .3878 .3378 .1678 .1177 .7877 .29
$7 / 678.0978 .2378 .3578 .4278 .3678 .4878 .4578 .3978 .4278 .3478 .3978 .478 .3978 .4178 .3878 .378 .2578 .2277 .9777 .33$
$4 / 378.2278 .2578 .1778 .0678 .0477 .8977 .9777 .9477 .9677 .8577 .9878 .0278 .1178 .1178 .1878 .1778 .2978 .2578 .2177 .74$
$\begin{array}{llllllllllllll}011 & 112 & 2^{13} & 3^{14} & 4^{5} & 516 & 61 & 7^{18} & 819 & 910 & 111^{10} & 10^{19} & 918 & 817 \\ 7^{16} & 615 & 51^{1 /} & 413 & 3^{12} & 2^{11}\end{array}$



Fig. 9: Classification accuracy(\%) patterns of all-class are exhibited for (a) $20 \times 20$ SIC models and (b) $19 \times 8$ convex focal losses. The corresponding histograms are shown in (c). In the case of the SIC model, the best test classification accuracy $78.56 \%$ is achieved when $\left(\alpha_{+}, \alpha_{-}\right)=(7 / 8,8 / 7)$. On the other hand, the convex focal loss model achieves the best test accuracy $78.39 \%$ at $(\pi, \gamma, \xi)=(0.5,3,0),(0.5,2,1)$. Moreover, a histogram comparison between the two models shows that more than half of all SIC models obtain at least $78.20 \%$ accuracy, while only $10 \%$ of $\pi$-weighted convex focal losses obtain that accuracy.
[29] L. A. Kurgan, K. J. Cios, R. Tadeusiewicz, M. Ogiela, L. Goodenday, "Knowledge discovery approach to automated cardiac SPECT diagnosis,", Artificial Intelligence in Medicine, vol. 23, pp. 149-169, 2001.
[30] J. Liao, O. Kosut, L. SANKAR, F. DU Pin Calmon, "Tunable measures for information leakage and applications to privacy-utility tradeoffs", IEEE Transactions on Information Theory, vol. 65, pp. 8043-8066, 2019.
[31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, pp. 318-327, 2020.
[32] Y. Lin, "Support Vector Machines and the Bayes rule in classification", Data Mining and Knowledge Discovery, vol. 6, pp. 259-275, 2002.
[33] H.-T. Lin, C.-J. Lin, R. C. WENG, "A note on Platt's probabilistic outputs for support vector machines", Mach. Learn., 68 (2007), pp. 267-276.
[34] K. LYU, Z. Li, R. WANG, AND S. Arora, "Gradient descent on two-layer nets: margin maximization and simplicity bias", Advances in Neural Information Processing Systems 35, 2021.
[35] A. F. T. Martins and R. F. Astudillo, "From softmax to sparsemax: A sparse model of attention and multi-label classification", Proceedings of the 33th International Conference on Machine Learning, 2016.
[36] H. MASNADI-Shirazi and N. VASCONCELOS, "On the design of loss functions for classification: theory, robustness to outliers, and savageboost", Advances in Neural Information Processing Systems 21, 2008.
[37] P. McCullagh and J. A. Nelder, Generalized Linear Models, Second Edition, Chapman \& Hall/CRC, 1989.
[38] J. J. Moré and D. J. Thuente, "Line search algorithm with guaranteed sufficient decrease", ACM Transactions on Mathematical Software vol. 20, pp. 286-307, 1994.
[39] K. P. Murphy, Machine Learning, MIT Press, 2012.
[40] M. Mutschler and A. ZEll, "Parabolic approximation line search for DNNs", Advances in Neural Information Processing Systems 33, 2020.
[41] J. Nocedal and S. J. Wright, Numerical Optimization, Second Edition, Springer-Verlag, 2006.
[42] K. Oksuz, B. C. Cam, S. Kalkan, and E. Akbas, "Imbalance problems in object detection: a review", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, pp. 3388-3415, 2021.
[43] Y. Ollivier, "Riemannian metrics for neural networks I: feedforward networks", Information and Inference: A Journal of the IMA, vol. 4. pp. 108-153, 2015.
[44] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods", in Advances in Large Margin Classifiers, A.J. Smola, P. Bartlett, B. Schölkopf, D, Schuurmans eds, MIT Press., 1999.
[45] M. D. Reid and R. C. Williamson, "Information, divergence and risk for binary experiments", Journal of Machine Learning Research, vol. 12, pp. 731-817, 2011.
[46] R. T. Rockafellar, Convex Analysis, Princeton University Press, Princeton, 1970.
[47] M. Schmidt, minFunc: unconstrained differentiable multivariate optimization in Matlab, http://www.cs.ubc.ca/ $\sim$ schmidtm/Software/ minFunc.html 2005.
[48] Wikipedia - sigmoid function https://en.wikipedia.org/wiki/Sigmoid_function
[49] D. Soudry, E. Hoffer, M. S. Nacson, S. Gunasekar, N. Srebro, "The implicit bias of gradient descent on separable data", Journal of Machine Learning Research, vol. 19, pp. 1-57, 2018.
[50] T. Sypherd, M. Diaz, J. K. Cava, G. Dasarathy, P. Kairouz, and L. Sankar, "A tunable loss function for robust classification: calibration, landscape, and generalization", IEEE Transactions on Information Theory, vol. 68, pp. 6021-6051, 2022.
[51] A. TSANAS AND A. XIFARA, "Accurate quantitative estimation of energy performance of residential building using statistical machine learning tools,", Energy Buildings, vol. 49, pp. 560-567, 2012.
[52] M. Kelly, R. Longjohn, and K. Nottingham, The UCI Machine Learning Repository, https://archive.ics.uci.edu
[53] G. VARDI, "On the implicit bias in deep-learning algorithms", Communications of the ACM, vol. 66, pp. 86-93, 2023.
[54] S. VASWANI, F. BACH, M. SChmidt, "Fast and faster convergence of SGD for over-parameterized models (and an accelerated Perceptron)", Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics, 2019.
[55] M. Wainberg, B. Alipanahi, and B. J. Frey, "Are random forests truly the best classifiers?", Journal of Machine Learning Research, vol. 17, pp. 1-5, 2016.
[56] R. Wedderburn, "Quasi-likelihood functions, generalized linear models, and the Gauss-Newton method", Biometrika, vol. 61, pp. 439-447, 1974.
[57] H. Woo, "A characterization of the domain of Beta-divergence and its connection to Bregman variational model", Entropy, vol. 19, 482, 2017.
[58] H. Woo, "Logitron: Perceptron-augmented classification model based on an extended logistic loss function", arXiv:1904.02958p1, 2019.
[59] H. Woo, "The Bregman-Tweedie classification model", arXiv: 1907.06923v1, 2019.
[60] H. Woo, "Bregman-divergence-guided Legendre exponential dispersion model with finite cumulants ( $k$-LED)", arXiv: 1910.03025v1, 2019.


[^0]:    H. Woo is with Logitron X, Republic of Korea, e-mail:hyenkyun@gmail.com.

