This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Ranking Regularization for Critical Rare Classes: Minimizing False Positives at a High True Positive Rate

Kiarash Mohammadi[1,2*] He Zhao[1] Mengyao Zhai[1] Frederick Tung[1]
[1]Borealis AI [2] Mila, Université de Montréal
kiarash.mohammadi@mila.quebec, {he.zhao, mengyao.zhai, frederick.tung}@borealisai.com

Abstract

*In many real-world settings, the critical class is rare and a missed detection carries a disproportionately high cost. For example, tumors are rare and a false negative diagnosis could have severe consequences on treatment outcomes; fraudulent banking transactions are rare and an undetected occurrence could result in significant losses or legal penalties. In such contexts, systems are often operated at a high true positive rate, which may require tolerating high false positives. In this paper, we present a novel approach to address the challenge of minimizing false positives for systems that need to operate at a high true positive rate. We propose a ranking-based regularization (**RankReg**) approach that is easy to implement, and show empirically that it not only effectively reduces false positives, but also complements conventional imbalanced learning losses. With this novel technique in hand, we conduct a series of experiments on three broadly explored datasets (CIFAR-10&100 and Melanoma) and show that our approach lifts the previous state-of-the-art performance by notable margins.*

1. Introduction

The cost of error is often asymmetric in real-world systems that involve rare classes or events. For example, in medical imaging, incorrectly diagnosing a tumor as benign (a false negative) could lead to cancer being detected later at a more advanced stage, when survival rates are much worse. This would be a higher cost of error than incorrectly diagnosing a benign tumour as potentially cancerous (a false positive). In banking, misclassifying a fraudulent transaction as legitimate may be more costly in terms of financial losses or legal penalties than misclassifying a legitimate transaction as fraudulent (a false positive). In both of these examples, the critical class is rare, and a missed detection carries a disproportionately high cost. In such situations, systems are often operated at high true positive rates, even though this may require tolerating high false positive rates. Unfortunately, false positives can undermine user confidence in the system and responding to them could incur other costs (e.g. additional medical imaging tests).

In this paper, we present a novel approach to address the challenge of minimizing false positives for systems that need to operate at a high true positive rate. Surprisingly, this

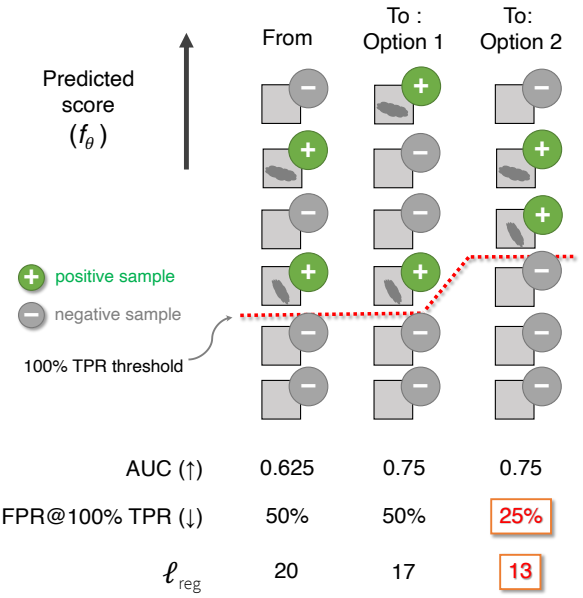

Figure 1. Which optimization option is preferred in operational contexts with critical positives? We propose a novel regularizer for systems that need to operate at a high true positive rate (TPR). Our approach prioritizes reducing false positives at a high TPR when presented with different options that equally improve the base objective, e.g. area under the ROC curve (AUC). In this toy example, option 2 is preferred because, with a suitable threshold depicted by the dashed line, all positives can be detected (100% TPR) with only one false positive (*i.e.*, 25% FPR at 100% TPR), better than option 1 where two false positives need be tolerated. Our regularizer is consistent with this preference: ℓ_{reg} is lower for option 2 than option 1 (*i.e.*, 13 *vs.* 17).

*Work done during an internship at Borealis AI

high-stakes operational setting has rarely been studied by the research community. In contrast to conventional imbalanced classification methods, we propose a general method for inducing a deep neural network to prioritize the reduction of false positives at a high true positive rate. To remain as broadly applicable as possible, we make minimal assumptions on the architecture and optimization details of the deep neural network. Our key insight is that the false positive rate at a high true positive rate is determined by how the least confident positives are ranked by the network. Our plug-and-play solution adds a simple yet effective ranking-based regularization term to the usual neural network training objective. The regularization term places an increasing penalty on positive samples the lower they are ranked in a sorted list of the network's classification scores, which works to push up the scores of the hardest positives.

Contributions. The main contributions of this paper are as follows:

- We present a novel plug-and-play regularization term that induces a deep neural network to prioritize the reduction of false positives in operational contexts where a high true positive rate is required.
- Our regularizer is generic and can be easily combined with other methods for imbalanced learning.
- We conduct extensive experiments on three public benchmarks to show how the proposed regularization term is complementary to conventional imbalanced learning losses, and achieves state-of-the-art performance in the high true positive rate operational setting.

2. Related work

Imbalanced classification. Class imbalance poses a challenge in training real-world neural network models. Without intervention, the optimization of a classification network becomes dominated by the common classes at the expense of the rare classes. Performing robust and efficient classification on imbalanced data is of practical importance and has seen much recent progress. In general, existing methods for imbalanced classification can be summarized into three major groups: cost function based, model output based, and data based. In the first group, various auxiliary optimization objectives are used during training, with a special attention to increase (or balance) the impact of under-represented classes. One such effort used weighted binary cross entropy (WBCE) [35], where the losses of minority samples are multiplied by a scaling factor (typically larger than one) to introduce more cost. Other cost function based approaches include symmetric margin loss (S-ML) [20], symmetric focal loss (S-FL) [18], asymmetric margin loss (A-ML) and focal loss (A-FL) [17], class-balanced BCE (CB-BCE) [8] and label distribution aware margin (LDAM) [4]. The second group of work tackles the problem by post-processing the model outputs. One simple approach divides the output scores with class co-occurrence frequencies [1, 15]. Similarly, other work adjusts the outputs by re-balancing the probabilities based on minority classes [28]. The third group of work focuses on data re-sampling or augmentation. Re-sampling or over-sampling the minority classes has shown solid benefits [6]. Enhancing samples from rare classes via augmentation also has received much consideration in recent years. Some representative examples are UnMix [31], ReMix [7] and MixUp [36]. Though approaching the same problem differently, all aforementioned methods have been used in a variety of vision tasks, *e.g.*, image classification [1,2,6,12], object detection [5,16] and image/video segmentation [14,32], where class imbalance is severe.

Our work falls into the first group (*i.e.* cost function based). We show in the experiments that our ranking-based regularization term is complementary to a wide range of established cost function based approaches (see Sec. 4). Importantly, in contrast to conventional imbalanced classification methods, we prioritize the reduction of false positives at a high true positive rate.

Differentiable ranking. Our regularization objective requires ranking the critical positive samples higher than the negative samples. This task is challenging as ranking operations are piece-wise constant functions, which have zero gradient almost everywhere [11]. Learning ranking directly through backprop is not feasible. To this end, many alternative solutions have been explored. For instance, an earlier effort considered using the expectation of the ranking as soft-ranker [27]. Some recent work used dynamic programming, instead of backprop, for weights update [26], while others achieved this purpose using a differentiable relaxation of histogram binning [3,13,21]. Other work has reformulated it as an optimal transport problem [9]. Our work relies on a recent advance that recasts the ranking operation as the minimizer of a linear combinatorial objective [29], which can be sovled elegantly by a blackbox combinatorial solvers [30] that can provide informative gradients from a continuous interpolation.

Deep AUC optimization. Our work can be considered (and will be demonstrated in Section 4) as an optimizer for maximizing the AUC score, *cf.* [10,19,33,34]. Nonetheless, none of them are designed to favor low false positive rates given a high true positive rate requirement. The work most similarly motivated to ours is ALM [24], which recently advocated yielding higher score (*i.e.* probability) for critical positives than non-critical negatives, thus improving the AUC. ALM formulates the problem from a constrained optimization perspective and achieves strong empirical results. We provide comprehensive comparisons with this state-of-the-art baseline in our experiments.

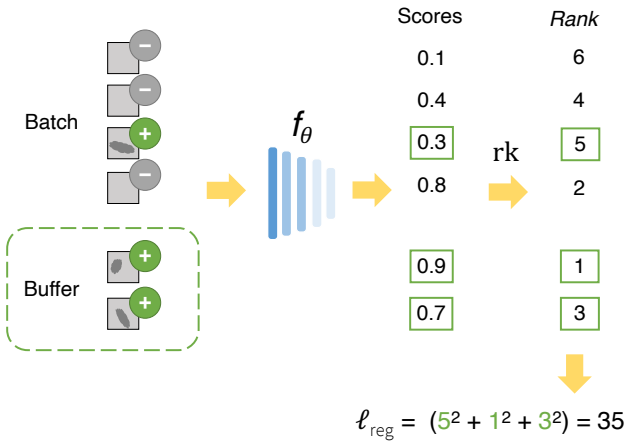

Figure 2. An illustration of the regularization term by example. Given a batch of inputs and a buffer (*i.e.*, external memory) of positive samples, we collect the probability scores from the classifier, f_θ, and apply the ranking function, $\boldsymbol{rk}(\cdot)$, to obtain a vector $\boldsymbol{r}$ of rank values. The rank values associated with the positive samples (*i.e.*, numbers in green boxes) are then used to compute the ranking-based regularization loss, ℓ_{reg}, using Eq. 4; we drop the normalization in this figure for simplicity of presentation. Note that f_θ can be an arbitrary classification architecture and RankReg does not rely on the specifics of that approach.

3. Technical approach

We present a novel, plug-and-play regularization loss as a generic method for inducing a neural network to prioritize minimizing false positives at a high true positive rate. First, we formulate the imbalanced binary classification task with critical positives in Sec. 3.1. Second, to obtain a solution that is tailored asymmetrically to the high true positive rate setting, we introduce a ranking-based regularization term that encourages models to rank the critical positives higher than the non-critical negatives, while prioritizing reducing the false positive rate at high true positive rate thresholds. Third, we discuss the used optimization solution in Sec. 3.4, as the ranking operation is challenging to optimize with back-propagation due to its non-differentiable nature.

3.1. Problem formulation

We seek to predict the correct label over a highly imbalanced binary classification dataset $D = \{(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_n, y_n), y_i \in \{0, 1\}\}$, where $\boldsymbol{x}_i$ are data samples, y_i are labels, and the critical data samples (positive class, labelled 1) appear much less frequently than the non-critical data samples (negative class, labelled 0). For example, medical images with cancerous tumors may be critical positives, while images with benign tumors or no tumors may be negatives. It is assumed that the cost of missing the positive class (a false negative) carries a disproportionately high cost, and that the system is required to operate at a high true positive rate.

Our goal is to produce a general method for inducing a deep neural network (DNN) classifier $f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}$, mapping d-dimensional inputs to output scores, to prioritize the reduction of false positives at a high true positive rate [24]. To be as general as possible, the method should make minimal assumptions on the architecture and optimization details of f_θ.

3.2. Preliminary

We propose a novel ranking-based [11, 29] regularizer that fulfills the above desiderata. Our key insight is that the false positive rate at a high true positive rate is determined by how the least confident positives are ranked by the network. Our plug-and-play approach adds a regularization term to the usual DNN training objective, making our solution complementary to a wide range of base objective functions, from conventional binary cross-entropy to more sophisticated imbalanced losses such as asymmetric focal loss [17]. In other words, we modify the DNN training objective to be:

$$\ell(f_\theta(\boldsymbol{x}), \boldsymbol{y}) = \ell_{base}(f_\theta(\boldsymbol{x}), \boldsymbol{y}) + \lambda\, \ell_{reg}(f_\theta(\boldsymbol{x}), \boldsymbol{y}), \quad (1)$$

where ℓ_{base} is the base objective function, ℓ_{reg} is the new regularization term, and λ is a balancing hyperparameter with value empirically set to 1.

3.3. Ranking regularizer

Denote by $\boldsymbol{rk}(\cdot)$ the ranking function that takes a vector of real values $\boldsymbol{a}$ and outputs the rank of each element in the sorted vector. In other words, the ith element in $\boldsymbol{rk}(\boldsymbol{a})$ is given by

$$\boldsymbol{rk}(\boldsymbol{a})_i = 1 + |\{j : \boldsymbol{a}_j > \boldsymbol{a}_i\}|. \quad (2)$$

Then, we devise a regularization term that is computed as the normalized sum over the squared rank values of the positive samples:

$$\boldsymbol{r} = \boldsymbol{rk}\left([f_\theta(\boldsymbol{x}_1), f_\theta(\boldsymbol{x}_2), \ldots, f_\theta(\boldsymbol{x}_B)]\right), \quad (3)$$

$$\ell_{reg}(f_\theta(\boldsymbol{x}), \boldsymbol{y}) = \frac{1}{|P|} \sum_{i=1}^{B} \boldsymbol{r}_i^2 \cdot \mathbb{1}[y_i = 1], \quad (4)$$

where $|P| = \sum_{i=1}^{B} \mathbb{1}[y_i = 1]$ is the number of positive samples in the batch of B samples. In practice, since positive samples may be severely under-represented in the dataset, we compute the regularization term over the union of the batch and an external memory that caches previous positive samples; we will revisit the implementation details of this buffer in Section 3.5. We also normalize the rank values $\boldsymbol{r}$ to be between 0 and 1.

To see how this regularization term prioritizes the reduction of false positives at a high true positive rate, let us consider the toy example in Figure 1. Suppose that the classifier f_θ currently produces the sorted ordering shown in the left column. The ground-truth critical positives are represented by green plus icons and the ground-truth negatives are represented by grey minus icons. The positive examples induce the second and fourth highest classification scores (higher is better for positives). To achieve a high true positive rate of 100% on these two positives, we would have to accept at least two false positives, obtaining a false positive rate of 50%. Now, suppose that in the next training iteration, the optimizer has two options, shown in the middle and right columns, that would equally improve the training objective; here, we illustrate with the area under the ROC curve (AUC), a common retrieval-based objective. While equally preferable by the training objective, the right column is better aligned with our goal of reducing false positives at a high true positive rate: with a suitable threshold (depicted by the dashed line), we can obtain a false positive rate of 33% at a true positive rate of 100%. On the other hand, the middle column can at best achieve a false positive rate of 50% at a true positive rate of 100%.

The proposed regularization term distinguishes between the middle and right columns, and assigns a higher loss to the middle column. In the middle column, the positives have the first and fourth highest classification scores, producing a regularization loss of $1^2 + 4^2 = 17$ (we drop the normalization terms here for simplicity of presentation). In the right column, the positives have the second and third highest classification scores, producing a regularization loss of $2^2 + 3^2 = 13$. The proposed regularization therefore favors the right column, as desired. Note that if we use the ranks directly instead of squaring, the regularization loss would be 5 in both cases ($1 + 4 = 2 + 3 = 5$). Squaring places an increasing penalty on positive samples the lower they are ranked in a sorted list of the network's output scores, which works to push up the scores of the least confident positive samples.

3.4. Optimization on ranking operations

Rank-based objectives often arise in computer vision [13, 25, 27] and are typically challenging to optimize due to the non-differentiability of the ranking function. The ranking function is piece-wise constant, i.e., perturbing the input would most likely not change the output. Thus, we cannot obtain informative gradients (i.e., gradients are zero almost everywhere). We adopt the optimization approach of [22], which frames the ranking function as a combinatorial solver and relies on an elegant way of backpropagating through blackbox combinatorial solvers [29]. The combinatorial objective version of computing the ranking function is given by

$$\boldsymbol{rk}(\boldsymbol{a}) = \underset{\boldsymbol{\pi} \in \Pi_n}{\arg\min}\, \boldsymbol{a} \cdot \boldsymbol{\pi}, \tag{5}$$

for an arbitrary vector $\boldsymbol{a}$ of real values (see Eq. 2), where Π_n is a set that contains all the permutations of $\{1, 2, \cdots, n\}$. This reframing enables us to leverage [29] to differentiate through a blackbox combinatorial solver: [29] proposes a family of piecewise affine continuous interpolation functions parameterized by a single hyperparameter that controls the tradeoff between faithfulness to the true function and informativeness of the gradient. In brief, we compute and return the gradient of the continuous interpolation: $\frac{\partial \mathcal{L}}{\partial \mathbf{a}} = -\frac{1}{\gamma}(\mathbf{rk}(\mathbf{a}) - \mathbf{rk}(\mathbf{a}_\gamma))$, where $\mathbf{a}_\gamma$ is a perturbed input derived from the incoming gradient information $\frac{\partial \mathcal{L}}{\partial \mathbf{rk}}$ via $\mathbf{a}_\gamma = \mathbf{a} + \gamma \frac{\partial \mathcal{L}}{\partial \mathbf{rk}}$. We refer the interested reader to [22] for further details.

3.5. Buffer of positive samples

During training, we maintain a buffer (*i.e.*, an external memory) of positive samples to enable the regularization term to be computed per batch even in datasets with severe imbalance ratios, as a batch may contain few (or no) positive samples. The whole buffer is always appended to the batch. We implement the buffer as a priority queue and it works as follows. At the start of training, positive samples are accumulated from the incoming batches and added to the buffer up to a fixed maximum capacity. Afterwards, as batches are processed, new positive samples replace the samples in the buffer for which the model is the most certain, *i.e.*, the buffered samples with the maximum f_θ responses. This replacement strategy keeps the hard positives in the buffer and removes positives for which the classifier is already confident. We consider the alternative replacement strategies of first-in-first-out (FIFO) and minimum f_θ responses in the ablation studies. The complete pipeline, including the buffer of positive samples, is illustrated by example in Figure 2 (without normalization for simplicity of presentation).

4. Empirical evaluation

4.1. Overview

To demonstrate the effectiveness of RankReg, extensive experiments are conducted on three public image-based benchmarks: binary imbalanced CIFAR-10, binary imbalanced CIFAR-100, and Melanoma. To adapt CIFAR-10 and CIFAR-100 to the critical positives setting, we follow the same experimental protocol as the state-of-the-art baseline [24]. As we do not have access to the private medical imaging dataset used in [24], we performed experiments on a publicly available medical imaging dataset (*i.e.*, Melanoma). In this section, we first describe the benchmarks, baselines, and training details. We then present ex-

Methods	Binary CIFAR10, imb. **1:100**			
	FPR@ ↓ 98%TPR	FPR@ ↓ 95%TPR	FPR@ ↓ 92%TPR	AUC ↑
BCE	56.0	45.0	29.0	91.2
+ALM	52.0	34.0	21.0	93.1
+RankReg	**47.1**	**26.2**	**20.6**	94.3
S-ML	59.0	40.0	26.0	91.7
+ALM	50.0	37.0	**24.0**	92.5
+RankReg	**45.6**	**31.4**	29.7	93.9
S-FL	59.0	40.0	27.0	91.7
+ALM	55.0	39.0	25.0	91.5
+RankReg	**53.3**	**35.4**	**20.7**	92.8
A-ML	54.0	36.0	23.0	92.4
+ALM	**45.0**	35.0	23.0	92.8
+RankReg	47.8	**28.9**	**21.4**	94.1
A-FL	50.0	38.0	24.0	92.3
+ALM	**49.0**	37.0	23.0	92.8
+RankReg	50.5	**28.7**	**20.9**	94.3
CB-BCE	89.0	72.0	59.0	78.0
+ALM	67.0	51.0	36.0	88.1
+RankReg	**48.8**	**29.9**	**24.6**	93.2
W-BCE	69.0	52.0	37.0	87.4
+ALM	66.0	48.0	31.0	89.3
+RankReg	**60.0**	**39.4**	**29.6**	92.1
LDAM	65.0	48.0	34.0	89.0
+ALM	60.0	42.0	31.0	91.0
+RankReg	**42.8**	**25.6**	**23.8**	95.0
Avg. Δ	6.0	9.7	2.8	2.3

Table 1. Comparison results for binary imbalanced CIFAR-10 showing FPRs at {98%, 95%, 92%} TPRs. Baseline numbers are quoted from ALM [24]. "+ALM" and "+RankReg" are shorthand for *BaseLoss*+ALM and *BaseLoss*+RankReg, respectively.

Methods	Binary CIFAR100, imb. **1:100**			
	FPR@ ↓ 98%TPR	FPR@ ↓ 95%TPR	FPR@ ↓ 90%TPR	AUC ↑
BCE	93.0	63.0	47.0	81.8
+ALM	91.0	49.0	39.0	82.7
+RankReg	**85.2**	**42.4**	**28.7**	85.5
S-ML	89.0	65.0	43.0	82.7
+ALM	88.0	69.0	41.0	81.7
+RankReg	**64.0**	**44.8**	**34.5**	85.4
S-FL	89.0	62.0	44.0	82.6
+ALM	88.0	60.0	42.0	81.7
+RankReg	**84.6**	**49.2**	**38.4**	84.7
A-ML	91.0	63.0	44.0	81.8
+ALM	89.0	55.0	37.0	82.7
+RankReg	**81.6**	**43.4**	**32.6**	85.5
A-FL	88.0	63.0	45.0	82.8
+ALM	86.0	62.0	40.0	83.2
+RankReg	**70.0**	**53.4**	**35.8**	84.6
CB-BCE	93.0	75.0	52.0	78.8
+ALM	**89.0**	59.0	36.0	83.8
+RankReg	89.8	**48.6**	**33.4**	84.1
W-BCE	88.0	59.0	41.0	79.7
+ALM	87.0	**53.0**	**39.0**	83.2
+RankReg	**84.0**	60.0	41.1	82.9
LDAM	84.0	70.0	42.0	82.3
+ALM	80.0	59.0	40.0	83.2
+RankReg	**70.3**	**51.6**	**35.0**	84.7
Avg. Δ	8.6	8.6	4.3	1.9

Table 2. Comparison results for binary imbalanced CIFAR-100 showing FPRs at {98%, 95%, 90%} TPRs. Baseline numbers are quoted from ALM [24]. "+ALM" and "+RankReg" are shorthand for *BaseLoss*+ALM and *BaseLoss*+RankReg, respectively.

perimental results on the three benchmarks, showing how the proposed regularizer is complementary to conventional imbalanced learning losses and achieves state-of-the-art results. We conclude with ablations and analyses of design choices.

4.2. Datasets and evaluation protocol

Binary imbalanced CIFARs. For a fair comparison, we adopt the same binary imbalanced versions of CIFAR-10 and 100 as curated by the authors of the state-of-the-art method [24]. In brief, binary imbalanced CIFAR-10 is constructed by randomly designating two classes as positives and negatives. All training samples from the negative class are used, while training samples from the positive class are subsampled. Binary imbalanced CIFAR-100 is constructed by designating one super-class as the negative class and a sub-class of a different super-class as the positive class. Again, all training samples from the negative class are used, while training samples from the positive class are subsampled. We refer the interested reader to [24] for the construction details. Following the evaluation protocol in [24], we experiment with 1:100 and 1:200 imbalance ratios (1 critical positive to 100 or 200 negatives), set aside 100 and 50 samples per class to form the validation set for hyperparameter selection for CIFAR-10 and 100 respectively, and evaluate on a class-balanced test set.

Melanoma. The Kaggle Melanoma dataset is a medical image classification dataset that was first proposed on a competition for identifying melanoma (a common form of skin cancer) in imaging scans of skin lesion [23]. It is composed of 33,126 images collected from patients in large variance, where only 584 out of the entire set are malignant (*i.e.*, positive) melanoma; therefore, the dataset has a 1:176 imbalance ratio. It is split into training, validation, and test sets

with ratios of 70%, 10%, and 20%, respectively. The original resolutions of images in Melanoma are too high (*e.g.*, [6000, 4000] or [1920, 1080]) to fit in the backbone network (*i.e.*, ResNet-18). We resize them into [256, 256] for the convenience of computational resources, *cf.* [34]. Since Melanoma is naturally imbalanced, no further curation is needed for our study.

Metrics. We evaluate the performance using the false positive rate (FPR) against several increasingly strict true positive rates (TPR), *i.e.*, FPR@$\{\beta\}$TPR and $\beta \in \{90\%, 92\%, 95\%, 98\%\}$. For completeness, we also evaluate using the area under curve metric (AUC) to reveal the overall classification performance, as typically seen in related work [10, 19, 33, 34].

Baseline methods. Following previous method [24], we consider applying our proposed regularizer with several different existing loss functions, most of which have been designed to handle class imbalance: binary cross-entropy (BCE), symmetric margin loss (S-ML) [20], symmetric focal loss (S-FL) [18], asymmetric margin loss (A-ML) and focal loss (A-FL) [17], cost-weighted BCE (WBCE) [35], class-balanced BCE (CB-BCE) [8], and label distribution aware margin (LDAM) [4].

4.3. Implementation details

Backbone architectures. For the binary imbalanced CIFAR datasets, we adopt the ResNet-10 architecture up to the second last layer as feature extractor and append a multi-layer perceptron (MLP) with shape [512$\rightarrow$ 2] as the classifier. For Melanoma, we adopt the richer architecture of ResNet-18 and repeat the same step to create the classifier.

Buffer usage. The buffer of positive samples can have a rebalancing effect when used to compute the base loss in addition to the usual batch samples. We leverage this effect when training models on the binary imbalanced CIFAR-10 and CIFAR-100, which have balanced test sets following the protocol in [24]. Since all splits in the Melanoma dataset follow the data's natural class imbalance, when training the Melanoma models we compute the base loss using the batch samples only. Throughout our experiments, we set the batch size to 64 when training other methods (including base methods and ALM). When training RankReg models, we use a buffer size of 32 and reduce the batch size to 32.

Hyper-parameter search. For CIFAR-10 and CIFAR-100, which we have in common with [24], we use the same hyper-parameters on all base loss functions, for the sake of fair comparison. For Melanoma dataset, we tune the hyper-parameters as follows. The more general parameters like learning rate and batch size are chosen and fixed to work with the BCE loss. For ALM [24], a two step grid-search is performed. In the first step, we perform a grid-search over ρ and $\mu^{(0)}$. We choose ρ from the set $\{2, 3\}$ and $\mu^{(0)}$ from the set $\{10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}, 10^{-7}, 10^{-8}, 10^{-9}\}$ (note that this is a slightly more thorough grid-search than the original paper). When these two are fixed, we search for the best δ from the set $\{0.1, 0.25, 0.5, 1.0\}$. These parameters are tuned based on the AUC on the validation set.

Methods	Melanoma, imb. **1:170**				
	FPR@ ↓ 98%TPR	FPR@ ↓ 95%TPR	FPR@ ↓ 92%TPR	FPR@ ↓ 90%TPR	AUC ↑
BCE	49.8	45.9	38.6	35.5	85.7
+ALM	49.9	41.8	40.0	37.7	85.6
+RankReg	**49.4**	**37.9**	**33.9**	**31.6**	**86.8**
S-ML	**46.6**	42.8	38.4	37.4	85.3
+ALM	51.3	**40.5**	39.8	36.2	83.5
+RankReg	54.6	42.4	**36.1**	**34.4**	**86.3**
S-FL	59.0	47.3	44.4	39.5	83.8
+ALM	**47.8**	42.7	39.2	38.1	84.0
+RankReg	56.6	**37.8**	**31.2**	**29.8**	**86.1**
A-ML	**47.5**	42.9	40.4	36.6	85.4
+ALM	51.0	41.5	37.5	37.1	83.7
+RankReg	58.3	**40.8**	**36.7**	**33.9**	**86.2**
A-FL	55.6	45.0	42.7	41.2	84.4
+ALM	49.0	42.4	40.1	38.1	83.6
+RankReg	**48.0**	**36.2**	**30.7**	**28.8**	**86.3**
CB-BCE	67.2	59.5	35.7	**33.2**	82.6
+ALM	60.8	59.5	46.3	45.8	81.5
+RankReg	**57.8**	**44.9**	35.7	34.7	**83.7**
W-BCE	69.0	52.0	37.0	32.1	87.4
+ALM	66.0	48.0	**31.0**	30.7	89.3
+RankReg	**56.4**	**41.1**	33.0	**30.5**	**90.9**
LDAM	**59.7**	48.2	46.2	**39.0**	**83.4**
+ALM	62.7	47.7	**43.3**	40.7	81.5
+RankReg	65.6	**47.5**	45.7	43.9	81.7

Table 3. Comparison results for Melanoma dataset showing FPRs at $\{98\%, 95\%, 92\%, 90\%\}$ TPRs. "+ALM" and "+RankReg" are shorthand for *BaseLoss*+ALM and *BaseLoss*+RankReg, respectively.

Model ensembling. For CIFAR-10 and CIFAR-100 where the datasets are rather small, we report results from 10 ensembling models for higher reliability and to diminish dataset-dependant biases, matching the protocol in [24]. In detail, 10 random stratified splits of the dataset are created and a model is trained on each. Finally, these models are ensembled by averaging their outputs in the logit space. We do not perform ensembling on Melanoma, as it is larger and has a standard data split.

4.4. Comparison to alternative approaches

Binary imbalanced CIFAR-10. Table 1 compares the performance of using RankReg as well as the previous state-of-the-art method ALM [24] together with eight base losses on CIFAR-10 dataset, curated with a imbalance ratio of 1:100 (see Sec. 4.2). We group the empirical results by base loss (BCE, S-ML, S-FL, etc.). Within each group, we first show the results obtained by applying the base method as well as

FPR@βTPR	CIFAR10				CIFAR100			
	98%	95%	92%	AUC	98%	95%	90%	AUC
Ranks	52.1	35.2	24.0	93.6	86.3	52.8	43.0	83.2
Squared ranks	47.1	26.2	20.6	94.3	85.2	42.4	28.7	85.5
Cubed ranks	45.5	31.9	23.0	93.7	84.2	53.8	50.4	83.7
Exponential of ranks	44.5	34.0	24.3	93.6	83.6	48.8	39.4	84.9

Table 4. Ablation study of different ranking penalty choices on imbalanced CIFAR-10 and 100 datasets.

FPR@βTPR	CIFAR100				Melanoma			
	98%	95%	92%	AUC	98%	95%	92%	AUC
Dequeue Max	85.2	42.4	28.7	85.5	49.4	37.9	33.9	86.8
FIFO	86.8	44.2	31.2	85.2	59.2	47.6	40.5	83.1
Dequeue Min	88.2	55.9	44.8	83.2				

Table 5. Ablation study on buffer update strategy. Swapping out the most confident sample with incoming ones (*i.e.*, Dequeue Max) performs better than other alternatives.

FPR@βTPR	CIFAR10				CIFAR100			
	98%	95%	92%	AUC	98%	95%	90%	AUC
Buffer = 0	58.8	43.6	30.2	90.4	93.4	48.4	37.4	83.1
Buffer = 5	53.0	42.4	28.7	92.6	86.6	57.2	39.0	82.5
Buffer = 10	48.1	26.9	26.2	94.5	85.2	50.2	28.7	84.6
Buffer = 20	47.6	26.2	22.0	93.8	85.2	50.0	29.8	84.9
Buffer = 32	47.1	26.2	20.6	94.3	85.2	42.4	28.7	85.5
Buffer = 48	46.1	24.3	23.2	93.9	85.2	42.1	27.6	85.5

Table 6. Illustration of the impact of the positive buffer. We report false positive rate results at high true positive rates for various buffer sizes.

the previous state-of-the-art approach. Then, we show our results. For each FPR and overall AUC, the best result is either underlined or highlighted in red text.

It is clear that our results are consistently better on most metrics, except for three FPR values at S-ML, A-FL and A-ML baselines, where RankReg is the second best approach. The performance improvement is especially striking when coupling RankReg with CB-BCE: RangReg reduces the FPR at the strictest TPR ratio by 18%, *i.e.*, from 67.0 to 48.8 in FPR@98%TPR. The best overall results are obtained by fusing RankReg with the LDAM baseline, where we achieve the higest AUC score (*i.e.*, 95.0) as well as the lowest FPR@98%TPR value (*i.e.*, 42.8) across all experimental results.

Even though our goal is *not* to have higher AUC scores, our approach obtains the new state-of-the-art AUC performance on all baselines. The bottom row in Table 1 shows the improvements on FPRs using RankReg compared to previous best results, averaged across all baselines. It is notable that our approach obtains larger gains at higher TPRs. For instance, our approach achieves 6.0 and 9.7 FPR improvements at 98% and 95% TPRs, respectively, compared to 2.8 at 92% TPR, which is favored by our goal (see Sec. 3.3).

Binary imbalanced CIFAR-100. To show the capability of our method to scale, we evaluate our method on the curated CIFAR-100 dataset. The results in Table 2 are consistent with our results on CIFAR-10. Once again, our approach is the top performer across most metrics. However, this time, both the BCE and A-ML baselines achieve the highest AUC score using RankReg. Moreover, it is notable that on the highest TPR (*i.e.*, 98%), our approach outperforms the previous state-of-the-art with the margin $> 10\%$ on 5 different baselines (*i.e.*, A-FL, S-ML, BCE, LDAM and S-FL). Such notable gains are only observed twice in previous experiments (*i.e.*, LDAM and CB-BCE in Table 1).

Melanoma. We demonstrate the application of RankReg on imbalanced cancer classification using the Melanoma benchmark and show results in Table 3. As we are the first to perform FPR vs. TPR study on such a large-scale dataset, there is a lack of comparison methods. Therefore, we provide results for all baselines as well as their combination with ALM by running experiments ourselves. It can be seen that, across all baselines, RankReg achieves state-of-the-art performance in the majority of metrics, with a minor setback on LDAM, where both ours and ALM achieve one best metric. These results verify the effectiveness of RankReg on a real-world dataset with critical positives.

Discussion. We attribute the performance lift obtained by RankReg to its more direct approach in ordering the critical positives ahead of the negatives, which is accomplished indirectly through margins in the previous state-of-the-art method ALM [24]. The FPR at a given TPR depends only on how samples are ranked relative to each other, and not on the magnitude of the classification scores. Furthermore, the FPR at a high TPR is determined by the ranking of the least confident positives. Our regularizer places an increasing penalty on positive samples the lower they are ranked, which works to push up the scores of the hardest positives.

4.5. Ablations and additional analyses

In this section, we examine the impact of used components as well as provide additional evaluation to reveal our system's pros and cons. For these additional studies, unless otherwise indicated, we present results from using our approach with BCE as the base function.

Rank penalty ablation. Table 4 shows an ablation study on different choices for the rank penalty in Eq. 4, including raw rank values $\boldsymbol{r}$, squared rank values $\boldsymbol{r}^2$, cubed rank values $\boldsymbol{r}^3$, and the exponential of rank values $e^{\boldsymbol{r}}$. Squared rank provides the best overall result while being simple.

Impact of buffer maintenance strategy. Our approach uses a buffer of critical positive samples to have meaningful ranking regularization signals at each batch of training. We evaluate the role of buffer by considering three kinds of maintenance strategies: (1) remove the most confident

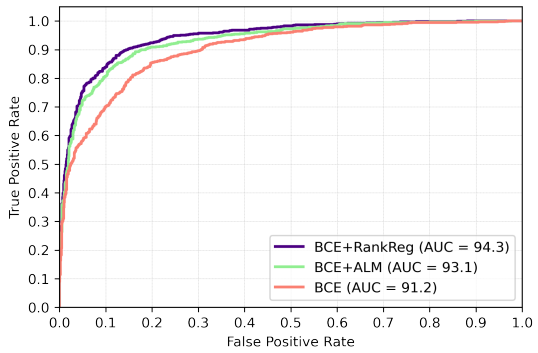

Figure 3. Evalution of ROC results produced by our approach *vs.* others on the CIFAR10 dataset (in 1:100 imbalance ratio).

sample while adding new positive samples from a incoming batch (*i.e.*, Dequeue Max), (2) first-in-first-out (*i.e.*, FIFO), and (3) remove the least confident sample (*i.e.*, Dequeue Min). The results in Table 5 show that feeding sufficient amount of low-ranking positives to the model is useful, as evidenced by the increased performance across all metrics.

Impact of buffer size. Throughout the empirical results in Sec. 4.4, we use a buffer size of 32. Next, we ablate the size of the buffer by allowing the model to use more buffered positive samples during training. Table 6 shows that the buffer plays an important role in our approach. Indeed, excluding the buffer component yields worse results; and performance (especially FPR@98TPR) improves quickly as buffer size increases. 32 seems to be an improving plateau.

Visualization of ROCs. To further estimate the effectiveness of our approach to reduce false positive rates at high true positive rates, we visualize our ROC curves as well as comparison methods, as shown in Figure 3. The top two curves (*i.e.*, ours and ALM) significantly surpass that of the BCE baseline on FPRs at earlier TPRs, *i.e.*, starting from 30% TPR and onward. Importantly, our approach performs on par with ALM up until $\sim$75% TPR, and then consistently yields lower FPR values ever since to almost 100% TPR.

Robustness to label noise. Real-world datasets often contain mislabeled data. To evaluate the robustness of our approach in the presence of label noise, we perform additional experiments in which we incrementally flip a proportion η of training labels. Figure 4 shows how FPR@{98, 95, 92}%TPR (left to right) degrade as a function of η in the range of $[0, 0.5]$, using BCE as base loss. These results suggest that RankReg is as robust to label noise as the state-of-the-art approach [24].

1:200 imbalance ratio. We also test our model on more imbalanced situations, *e.g.*, 1:200 imbalance ratio. To this end, we use the same data curation pipeline as introduced in Sec. 4.2 and build binary imbalanced CIFAR-10 and 100 datasets with a 1:200 imbalance ratio. We defer the full result tables to the supplementary material for space reasons. Looking at the averaged improvements (*i.e.*, Avg.Δ) in the bottom row, our approach leads by a large margin.

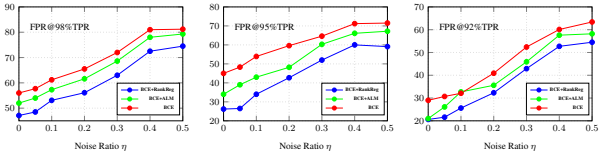

Figure 4. Label noise experiments using BCE as base loss on CIFAR10. We report FPR@{98, 95, 92}%TPR (left to right) with varied noise ratios.

Error@β%TPR $\downarrow$	**LT-CIFAR10** imb. 100			**LT-CIFAR10** imb. 200		
	80%	90%	Acc.	80%	90%	Acc
CE	29.8	34.7	70.4	37.8	42.4	64.0
CE+ALM	28.9	33.9	70.9	36.1	39.9	65.1
CE+RankReg	26.7	29.3	71.6	36.7	37.8	65.0

Table 7. Multi-class experiments using long-tailed CIFAR-10. Baseline numbers are quoted from ALM [24].

Multi-class extension. RankReg can be used in multi-class settings by ranking the critical samples higher than others based on the output probability for each class. Table 7 shows additional results in the multi-class setting using long-tailed CIFAR-10 following the experiment protocol in [24]. We report the average error rate of other classes after setting thresholds for {80, 90}%TPR on the critical class [24]. Our method performs better than [24] under the 1:100 imbalance ratio setting and comparably under the 1:200 setting.

5. Conclusion

The problem setting of critical rare positives has been surprisingly under-studied in the research community. This paper introduces a general method for inducing a neural network to prioritize the reduction of false positives when the operational context calls for a high true positive rate. Motivated by the observation that the false positive rate at a high true positive rate is determined by how the least confident positives are ranked by the network, we formulated a ranking-based regularizer that places an increasing penalty on positive samples the lower they are ranked in a sorted list of the network's classification scores. Experimental results show how our regularizer can be combined with a wide range of conventional losses and achieves state-of-the-art results in standard evaluations. We hope that our findings will inspire broader interest in this important problem setting, as well as provide practitioners a simple yet effective method to train better neural network models for critical rare classes.

References

[1] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural networks*, 106:249–259, 2018. 2

[2] Jonathon Byrd and Zachary Chase Lipton. What is the effect of importance weighting in deep learning? In *ICML*, 2019. 2

[3] Fatih Cakir, Kun He, Xide Xia, Brian Kulis, and Stan Sclaroff. Deep metric learning to rank. In *CVPR*, 2019. 2

[4] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *NeurIPS*, 2019. 2, 6

[5] Nadine Chang, Zhiding Yu, Yu-Xiong Wang, Anima Anandkumar, Sanja Fidler, and Jose M Alvarez. Image-level or object-level? A tale of two resampling strategies for long-tailed detection. In *ICML*, 2021. 2

[6] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002. 2

[7] Hsin-Ping Chou, Shih-Chieh Chang, Jia-Yu Pan, Wei Wei, and Da-Cheng Juan. Remix: Rebalanced mixup. In *ECCVW*, 2020. 2

[8] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, 2019. 2, 6

[9] Marco Cuturi, Olivier Teboul, and Jean-Philippe Vert. Differentiable ranks and sorting using optimal transport. In *NeurIPS*, 2019. 2

[10] Wei Gao and Zhi-Hua Zhou. On the consistency of AUC pairwise optimization. In *IJCAI*, 2015. 2, 6

[11] Yu Gong, Greg Mori, and Frederick Tung. Ranksim: Ranking similarity regularization for deep imbalanced regression. In *ICML*, 2022. 2, 3

[12] Hui Han, Wenyuan Wang, and Binghuan Mao. Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning. In *ICIC*, 2005. 2

[13] Kun He, Fatih Cakir, Sarah Adel Bargal, and Stan Sclaroff. Hashing as tie-aware learning to rank. In *CVPR*, 2018. 2, 4

[14] Shruti Jadon. A survey of loss functions for semantic segmentation. In *CIBCB*, 2020. 2

[15] Steve Lawrence, Ian Burns, Andrew Back, Ah Chung Tsoi, and C Lee Giles. Neural network classification and prior class probabilities. In *Neural networks: tricks of the trade*. Springer, 1998. 2

[16] Bo Li, Yongqiang Yao, Jingru Tan, Gang Zhang, Fengwei Yu, Jianwei Lu, and Ye Luo. Equalized focal loss for dense long-tailed object detection. In *CVPR*, 2022. 2

[17] Zeju Li, Konstantinos Kamnitsas, and Ben Glocker. Overfitting of neural nets under class imbalance: Analysis and improvements for segmentation. In *MICCAI*, 2019. 2, 3, 6

[18] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *CVPR*, 2017. 2, 6

[19] Mingrui Liu, Zhuoning Yuan, Yiming Ying, and Tianbao Yang. Stochastic AUC maximization with deep neural networks. *arXiv preprint arXiv:1908.10831*, 2019. 2, 6

[20] Weiyang Liu, Yandong Wen, Zhiding Yu, and Meng Yang. Large-margin Softmax loss for convolutional neural networks. In *ICML*, 2016. 2, 6

[21] Jerome Revaud, Jon Almazan, Rafael Sampaio de Rezende, and Cesar Roberto de Souza. Learning with average precision: Training image retrieval with a listwise loss. In *ICCV*, 2019. 2

[22] Michal Rolínek, Vít Musil, Anselm Paulus, Marin Vlastelica, Claudio Michaelis, and Georg Martius. Optimizing rank-based metrics with blackbox differentiation. In *CVPR*, 2020. 4

[23] Veronica Rotemberg, Nicholas Kurtansky, Brigid Betz-Stablein, Liam Caffery, Emmanouil Chousakos, Noel Codella, Marc Combalia, Stephen Dusza, Pascale Guitera, David Gutman, Allan Halpern, Harald Kittler, Kivanc Kose, Steve Langer, Konstantinos Lioprys, Josep Malvehy, Shenara Musthaq, Jabpani Nanda, Ofer Reiter, George Shih, Alexander Stratigos, Philipp Tschandl, Jochen Weber, and H. Peter Soyer. https://arxiv.org/abs/2008.07360. 5

[24] Sara Sangalli, Ertunc Erdil, Andreas Hoetker, Olivio Donati, and Ender Konukoglu. Constrained optimization to train neural networks on critical and under-represented classes. In *NeurIPS*, 2021. 2, 3, 4, 5, 6, 7, 8

[25] Yang Song, Alexander Schwing, Raquel Urtasun, et al. Training deep neural networks via direct loss minimization. In *ICML*, 2016. 4

[26] Yang Song, Alexander G. Schwing, Richard S. Zemel, and Raquel Urtasun. Training deep neural networks via direct loss minimization. In *ICML*, 2016. 2

[27] Michael Taylor, John Guiver, Stephen Robertson, and Tom Minka. SoftRank: Optimizing non-smooth rank metrics. In *ICWDM*, 2008. 2, 4

[28] Junjiao Tian, Yen-Cheng Liu, Nathaniel Glaser, Yen-Chang Hsu, and Zsolt Kira. Posterior re-calibration for imbalanced datasets. In *NeurIPS*, 2020. 2

[29] Marin Vlastelica, Anselm Paulus, Vít Musil, Georg Martius, and Michal Rolínek. Differentiation of blackbox combinatorial solvers. In *ICLR*, 2020. 2, 3, 4

[30] Marin Vlastelica, Anselm Paulus, Vít Musil, Georg Martius, and Michal Rolínek. Differentiation of blackbox combinatorial solvers. In *ICLR*, 2020. 2

[31] Zhengzhuo Xu, Zenghao Chai, and Chun Yuan. Towards calibrated model for long-tailed visual recognition from prior perspective. In *NeurIPS*, 2021. 2

[32] Michael Yeung, Evis Sala, Carola-Bibiane Schönlieb, and Leonardo Rundo. Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation. *Computerized Medical Imaging and Graphics*, 95:102026, 2022. 2

[33] Yiming Ying, Longyin Wen, and Siwei Lyu. Stochastic online AUC maximization. In *NeurIPS*, 2016. 2, 6

[34] Zhuoning Yuan, Yan Yan, Milan Sonka, and Tianbao Yang. Large-scale robust deep AUC maximization: A new surrogate loss and empirical studies on medical image classification. In *CVPR*, 2021. 2, 6

[35] Bianca Zadrozny, John Langford, and Naoki Abe. Cost-sensitive learning by cost-proportionate example weighting. In *ICDM*, 2003. 2, 6

[36] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. 2