Multiclass Minimax Learning for Deep Neural Networks

Cyprien GILET
Université de Technologie de Compiègne
CNRS, Heudiasyc Laboratory
Compiègne, France
cyprien.gilet@hds.utc.fr

Marie Guyomard
Université Côte d'Azur
CNRS, I3S
Sophia-Antipolis, France
guyomard@i3s.unice.fr

Susana Barbosa
Université Côte d'Azur
CNRS, IPMC
Sophia-Antipolis, France
sudocarmo@gmail.com

Lionel Fillatre
Université Côte d'Azur
CNRS, I3S
Sophia-Antipolis, France
fillatre@i3s.unice.fr

***Abstract*—For classification tasks, deep neural networks seek to minimize the average risk of classification error during the training step. When some classes are more difficult to recognize, the class-conditional classification probabilities, also called the recalls, of the neural network classifier are generally very unequal in many real world applications. This paper proposes a multiclass minimax approach for equalizing the recalls of a deep neural network. Our approach replaces the top layer of the neural network by a specific discrete minimax decision rule. This novel top layer is based on a K-means partitionning of its input deep features to train a minimax Bayes classifier that can fit any input statistical distribution. The learning process, based on a subgradient optimization algorithm, is scalable when the number of classes is large. Numerical experiments compare our approach to several state-of-the-art algorithms on medical images and CIFAR-100 database.**

***Index Terms*—Deep Neural Network, Minimax Learning, Multiclass Recall Equalization, Scalability.**

I. Introduction

Deep Neural Networks (DNNs) become unavoidable for classifying signals and images [1], [2] as they generally allow to achieve high classification performance. Deep neural networks offer a powerful representation mechanism. They produce features that contain useful information about the classification task they adress. But the control of classification errors is extremely difficult with DNNs. DNNs often suffer when the number of classes to recognize is greater than two. Generally, some classes are more difficult to classify than others and they will have a smaller recall, .i.e., the fraction of class-conditional instances that were correctly classified is small. This issue about unequal recalls occurs in many fields: imbalanced datasets [3], [4], [5], [6], [7], [8] when the class proportions are different, prior probability shifts [9] when the class proportions between the training set and the test set are different, robustness to adversarial attacks [10], [11] and so on. Equalizing the class recalls is therefore essential to make DNNs robust.

The minimax criterion [12], [13], [14], [15] is well known for identifying the most difficult classes and equalizing as well as possible the class-conditional classification probabilities [9], [16]. The minimax approach has often been applied to machine learning as in [17], [18], [19], [20], [21], [22]. It is more rarely applied to DNNs. The first attempt seems to be [16] where the authors proposed a fixed-point algorithm that requires to resample the training dataset at each iteration. A more general study is proposed in [23] where the data distribution can vary within a predefined bounded set. Recently, the Deep Minimax Probability Machine [11] applies minimax probability machine [19], [20] to DNNs in an end-to-end fashion. The authors put the minimax probability machine on top of a DNN and, instead of maximizing the likelihood of labels for data, they employ the objective function of minimax probability machine. The Deep Minimax Probability Machine is limited to only two-category classification tasks. Furthermore, it is just an approximation of the minimax Bayes classifier. The Bayes optimality is only achieved under some conditions (generally as the number of the training samples is infinitely large) provided in [24]. Contrary to the Deep Minimax Probability Machine, our new approach can deal with several classes and the Bayes optimality is guaranteed non-asymptotically for discretized features. Let us note that all these works, including our paper, should not be confused with the DNN learning rate study in the minimax sense as in [25], [26]. The minimax learning rate study is focused on the loss optimization rate, not on the calculation of a minimax classifier.

In this paper, we combine the deep neural network features and a Bayes classifier that is trained to be a minimax classifier. Our approach assumes that the DNN is first trained with the usual softmax layer. Then, our coupling method makes the DNN robust by replacing the softmax with a classifier that takes into account the feature global information and learns the worst-case probability of misclassification. We can interpret our model as applying the minimax Bayes classifier to the final hidden layer of a DNN, instead of using a softmax layer, as shown in Figure 1. The minimax classifier coincides with the Bayes classifier that maximizes the probability of misclassification or, equivalently, minimizes the accuracy.

The main novelties of our classifier are the following. First, in order to be versatile, our classifier discretizes the features that were the inputs of the softmax layer. By this way, the joint distribution of the discretized features is described by a probability mass function. Our classifier can be applied to any probability mass function. Any discretization can be suitable

Thanks to Fondation FondaMental, the Provence-Alpes-Côte d'Azur region, EUR DS4H and the SAFE AI Chair for funding.

but the K-means algorithm is a very satisfying discretization method. By controlling the number of K-means discretization values, we can deal with any input feature dimensions. Next, we propose a closed-form of the classification risk within a Bayesian framework. To get the minimax classifier, we must compute the worst-case class probabilities, i.e., the prior probality of each class such that the Bayes risk is maximum. Finally, a subgradient algorithm is proposed to maximize the Bayes risk. This optimization step is proved to converge and can deal with a large number of classes.

The paper is organized as follows. Section II shows how to couple a DNN with a minimax classifier. We partition the deep features computed by the DNN with a multivariate discretizer to get a closed-form empirical Bayes risk. We then maximize this risk with a projected-subgradient algorithm that computes the minimax classifier. Section III illustrates the benefits of our approach on medical images and the CIFAR-100 database. We compare our algorithm to other kind of machine learning classifiers (SVM, KNN, etc.) applied on features produced by a backbone DNN like ResNet-18 and EfficientNet-B7. The choice of the backbone DNN does not matter because our approach can fit any DNN. Experimental results demonstrate an encouraging performance. Section IV concludes the paper.

II. Coupling DNN with Minimax Learning

A. Coupling a trained DNN with an output classifier

Let $\mathcal{Y} = \{1, \ldots, K\}$ be the set of $K \geq 2$ class labels. Let $\Phi : \mathcal{X} \to \mathcal{Y}$ be a DNN [27] which assigns a class label to each signal or image X in the set $\mathcal{X}$. The architecture Φ is composed of s hidden layers $h_1, \ldots, h_s$ modeled as

$$\Phi(X) \;=\; h_{s+1} \circ h_s \circ \cdots \circ h_1(X) \;=\; h_{s+1} \circ \varphi(X), \quad (1)$$

where $h_{s+1}(\cdot)$ denotes the output layer, $\varphi(X)$ is the output of the last hidden layer and $f \circ g(X) = f(g(X))$ denotes the composition of functions. In the rest of the paper, $Z = \varphi(X) \in \mathbb{R}^d$ is called the deep features and h_{s+1} is called the output classifier layer. Usually in a DNN, the output decision rule h_{s+1} aims to approximate the Bayes classifier. The softmax classifier [27] is generally used to carry out this approximation.

This paper proposes to replace the output classifier layer with a minimax decision rule. Thus, we are studying DNNs that can be expressed as

$$\Phi_\delta(X) \;=\; \delta \circ \varphi(X) \;=\; \delta(Z), \quad (2)$$

where $\delta : \mathbb{R}^d \to \mathcal{Y}$ is any decision rule playing the role of the output classifier. In other words, $\Phi_\delta(X)$ is a DNN that takes a decision based on the deep features Z. We do not want to train again the hidden layers of the DNN but just to couple the deep features with a specific classifier (only this classifier will be trained). Our approach is a kind of fine tuning [28]. Let $\Delta := \{\delta : \mathbb{R}^d \to \mathcal{Y}\}$ denote the set of all output classifiers.

B. Empirical risk of the coupled DNN

Let $\mathcal{S} = \{(Y_i, X_i)\,, i \in \mathcal{I}\}$ be the training dataset containing m labeled training signals/images, where $\mathcal{I}$ is a finite set of indices. Let $L : \mathcal{Y} \times \mathcal{Y} \to [0, +\infty)$ be the loss function such that $L(k, l) := L_{k,l}$ is the loss of predicting the class l when the actual class is k. The empirical risk of the DNN Φ_δ is

$$\hat{r}\,(\Phi_\delta) = \frac{1}{m} \sum_{i \in \mathcal{I}} L(Y_i, \Phi_\delta(X_i)) = \frac{1}{m} \sum_{i \in \mathcal{I}} L(Y_i, \delta(Z_i)), \quad (3)$$

where $Z_i = \varphi(X_i)$. Hence, every DNNs of the form $\Phi_\delta(X)$ can be compared by evaluating only the risk $\hat{r}_\varphi\,(\delta)$ defined by

$$\hat{r}_\varphi\,(\delta) \;:=\; \hat{r}\,(\delta \circ \varphi) \;=\; \frac{1}{m} \textstyle\sum_{i \in \mathcal{I}} L(Y_i, \delta(Z_i)). \quad (4)$$

The empirical risk $\hat{r}\,(\Phi_\delta)$ of a DNN Φ_δ is equal to the empirical risk $\hat{r}_\varphi\,(\delta)$ of the rule δ applied on the deep features.

Let $\pi := [\pi_1, \ldots, \pi_K]$ be the class proportions of the training set such that π_k is the proportion of signals/images in class k. The empirical risk $\hat{r}_\varphi\,(\delta)$ can be written as [9], [29]

$$\hat{r}_\varphi\,(\delta) = \textstyle\sum_{k \in \mathcal{Y}} \pi_k \hat{R}_k\,(\delta)\,, \quad (5)$$

where $\hat{R}_k\,(\delta)$ is the empirical class-conditional risk of δ

$$\hat{R}_k\,(\delta) \;=\; \textstyle\sum_{l \in \mathcal{Y}} L_{k,l}\, \hat{\mathbb{P}}(\delta \circ \varphi(X_i) = l \mid Y_i = k), \quad (6)$$

and $\hat{\mathbb{P}}(\cdot \mid \cdot)$ is the conditional probability estimated from $\mathcal{S}$. The minimax criterion minimizes $M(\delta) = \max_k \hat{R}_k\,(\delta)$.

C. Minimax learning

The calculation of the empirical Bayes risk

$$\hat{r}_\varphi^B = \min_{\delta \in \Delta} \hat{r}_\varphi\,(\delta) \quad (7)$$

is intractable because we can not fit a well estimated empirical probability distribution on the deep features: their dimension is large and the number of samples is generally limited. Hence, we propose to discretize with a multivariate quantizer the deep features Z and to learn the minimax classifier by using a closed-form expression of $\hat{r}_\varphi^B$. Some papers show that the feature partitioning can improve significantly the performance of the classifier [30], [31], [32], [33]. Our approach is summarized on Fig. 1. The steps are described hereafter.

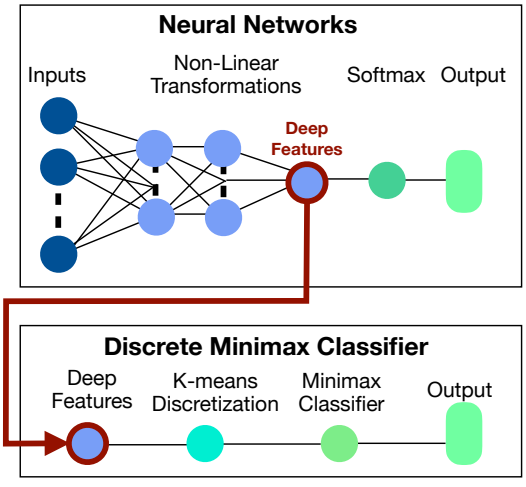

Fig. 1. Schema of our coupling method.

TABLE I
OVERVIEW OF EACH DATABASE: m^{train}, m^{val}, m^{test} CORRESPOND RESPECTIVELY TO THE NUMBER OF IMAGES IN THE TRAINING, THE VALIDATION AND THE TEST SETS, AND MIN, RESP. MAX, DENOTES THE MINIMUM, RESP. MAXIMUM, OF THE CLASS PROPORTIONS.

Database	K	m^{train}	m^{val}	m^{test}	π^{train}	π^{val}	π^{test}
DermaMNIST	7	7,007	1,003	2,005	Min = 0.01 Max = 0. 67	Min = 0.01 Max = 0.67	Min = 0.01 Max = 0.67
BreastMNIST	2	4,709	524	624	Min = 0.27 Max = 0.73	Min = 0.27 Max = 0.73	Min = 0.27 Max = 0.73
OCTMNIST	4	97,477	10,832	1,000	Min = 0.08 Max = 0.47	Min = 0.08 Max = 0.47	Min = 0.25 Max = 0.25

K-means partitioning. We have tested different methods of multivariate discretization but the K-means algorithm is the best trade-off between simplicity and efficiency. Hence, the deep feature space $\mathbb{R}^d$ is partitioned into T disjoint regions $\{\Omega_1, \ldots, \Omega_T\}$ such that $\cup_{t=1}^{T} \Omega_t = \mathbb{R}^d$. This defines a mapping $\gamma : \mathbb{R}^d \mapsto \mathbb{T} = \{1, \ldots, T\}$ such that $\gamma(Z) = t$ if and only if $Z \in \Omega_t$. Because a too large value of T can lead to overfitting regarding the average risk (4), T is chosen by cross-validation. We test several values of T and we compare the training error with the validation error of the minimax classifier for each value of T. We keep the value T such that the validation error is minimum while the gap, say ε_T, between the training and validation errors remains reasonable, about 10% for example. From the K-means partitioning, we obtain the labeled learning instances $\mathcal{S}_\gamma = \{(Y_i, t_i), i \in \mathcal{I}\}$ where $t_i = \gamma(\varphi(X_i))$ is the discrete deep feature profile of the initial signal/image X_i. Since the deep feature space is now partitioned, we limit the class of classifiers Δ to the class of classifiers Δ_γ:

$$\Delta_\gamma = \{\delta_\theta : \delta_\theta(Z) = \theta(\gamma(Z)) \text{ with } \theta : \mathbb{T} \to \mathcal{Y}\}, \quad (8)$$

where $\theta(\cdot)$ denotes any classifier from the discrete space $\mathbb{T}$ into the set of labels $\mathcal{Y}$. A classifier $\delta_\theta(\cdot)$ splits the whole space $\mathbb{R}^d$ into T classification regions. It follows that $\hat{r}_\varphi(\delta)$ in (4) is well approximated by

$$\hat{r}_\gamma(\theta) := \hat{r}(\theta \circ \gamma \circ \varphi) = \frac{1}{m} \sum_{i \in \mathcal{I}} L(Y_i, \delta_\theta(Z_i)) \quad (9)$$

where our attention is focused on classifiers $\delta_\theta \in \Delta_\gamma$.

Minimax classifier. The optimal Bayes classifier associated to the prior π and that minimizes $\hat{r}_\gamma(\theta)$ is

$$\theta_\pi^B := \arg\min_{\theta \in \Delta_\theta} \hat{r}_\gamma(\theta), \quad \text{where } \Delta_\theta = \{\theta : \mathbb{T} \to \mathcal{Y}\}. \quad (10)$$

The classifier θ_π^B is optimal only for the priors π. As shown in [9], [29], the minimax decision rule $\theta_{\bar{\pi}}^B$ is given by the Bayes classifier associated with the priors $\bar{\pi}$ maximizing $\hat{r}_\gamma\left(\theta_\pi^B\right)$ with respect to π. Thanks to the multivariate partitioning, the Bayes risk $\hat{r}_\gamma(\theta_\pi^B)$ can be explicitly calculated as

$$\hat{r}_\gamma(\theta_\pi^B) = \sum_{t=1}^{T} \min_{1 \le q \le K} \lambda_{q,t} \text{ with } \lambda_{q,t} = \sum_{k=1}^{K} L_{k,q}\, \pi_k\, \hat{p}_{k,t}, \quad (11)$$

where $\hat{p}_{k,t}$ denotes the probability of observing the discrete deep feature profile $\gamma(Z) = t$ given that the class label is k:

$$\hat{p}_{k,t} := \frac{|(Y_i, t_i) \in \mathcal{S}_\gamma : t_i = t, Y_i = k|}{|(Y_i, t_i) \in \mathcal{S}_\gamma : t_i = t|}, \quad (12)$$

where $|A|$ is the number of elements of the set A. The value $\lambda_{q,t}$ represents the average loss to decide the class q when we observe the discrete profile t. The function $\hat{r}_\gamma(\theta_\pi^B)$ is concave and piecewise affine over the simplex $\mathbb{S} = \{\pi \in [0,1]^K : \sum_{k=1}^{K} \pi_k = 1\}$ but not differentiable with respect to π because of the minimum function over q. In order to compute the value $\bar{\pi} = \text{argmax}_{\pi \in \mathbb{S}} \hat{r}_\gamma(\theta_\pi^B)$, we refer to the projected sub-gradient algorithm used in [9] which is proved to converge very accurately to $\bar{\pi}$.

Output. The final output layer of our approach is the minimax classifier $\delta_{\bar{\pi}}^B$ that is explicitly given by

$$\delta_{\bar{\pi}}^B(Z_i) = \theta_{\bar{\pi}}^B(t_i) = \arg\min_{1 \le q \le K} \lambda_{q,t_i} \text{ with } t_i = \gamma(Z_i). \quad (13)$$

This classifier is expressed in a closed-form and is easy to use in practice. Furthermore, our approach is scalable since i) the K-means clustering allows us to control the dimension T and ii) the sub-gradient algorithm works well in high dimensions.

III. EXPERIMENTS

Medical databases. We consider three real medical databases [34] which differ according to the number of images, the number of classes and the class proportions (see Table I). As shown in Fig. 2, DermaMNIST corresponds to dermatoscopic images of common pigmented skin lesions, OCTMNIST to optical coherence tomography images for retinal diseases, and BreastMNIST to breast ultrasound images for which the objective is to classify benign and malignant tumors [34]. Each database contains a training set, a validation set and a test set with 28×28 pixel images.

To illustrate that our approach can be coupled with any kind of DNN, we considered two Convolutional Neural Networks (CNN): *ResNet-18* [35] and *EfficientNet-B7* [36]. We trained each CNN on the training set with 100 epochs using the cross-entropy loss and a SGD optimizer as in [37]. We compared six multiclass classifiers that exploit the deep features: the softmax classifier which was considered in the initial CNN, the Discrete Bayes Classifier (DBC) [15], the K-Nearest Neighbors (KNN), the Support Vector Machine (SVM), the Weighted Random Forests (WRF) and our Discrete Minimax Classifier (DMC). Each output classifier was fitted on the deep features associated with the validation set in order to avoid overfitting regarding the average risk (4) possibly due to the deep features coming from the training set. The hyperparameters of DBC, KNN,

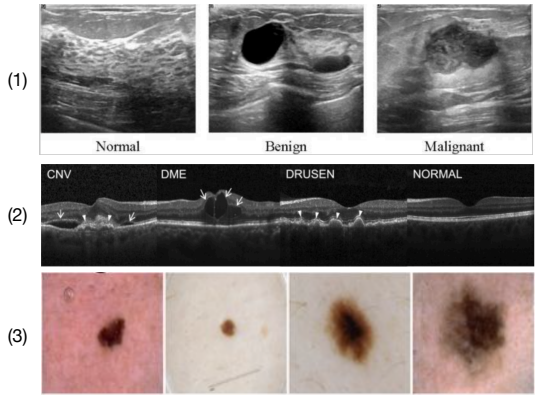

Fig. 2. (1) BreastMNIST, (2) OCTMNIST, (3) DermaMNIST.

WRF and DMC were tuned from a 4-folds cross-validation procedure on the validation set. More precisely, the optimal number T of discrete profiles regarding the DMC was chosen as explained in the Kmeans partitioning paragraph with for example a maximum gap $\varepsilon_T = 9\%$ on the BreastMinist database. The generalization performance of each output classifier was then evaluated on the test set.

Table II compares the results on the validation and test sets of each output classifier with respect to three criteria: the average risk $\hat{r}_\varphi(\delta)$ defined in (4), the maximum of the class-conditional risks, and the difference between the maximum and the minimum class-conditional risk: $\psi(\delta) := \max_{k\in\mathcal{Y}} \hat{R}_k(\delta) - \min_{k\in\mathcal{Y}} \hat{R}_k(\delta)$. For these experiments, we considered the $L_{0\text{-}1}$ loss function defined by $L_{kk} = 0$ and $L_{kl} = 1$ when $k \neq l$, so that the usual accuracy of a DNN is equal to $1 - \hat{r}_\varphi(\delta)$ and the minimum recall is equal to $1 - \max_{k\in\mathcal{Y}} \hat{R}_k(\delta)$.

The conclusions are the following. First, the DBC, KNN and SVM output classifiers generally yield to similar results as the initial CNN using the softmax output layer. This illustrates that all these decision rules tend to converge to the Bayes classifier. However, the maximum class-conditional risks associated with these output classifiers always appear too high. In other words, these classifiers do not provide efficient predictions for the difficult classes with the smallest number of images, even if these classes correspond to diseases. When we need to well classify the most difficult classes and to balance the risks per class, a trade-off is necessary: allowing the global risk $\hat{r}_\varphi(\delta)$ to become higher in order to better classify the difficult classes. This is confirmed by WRF (which are generally known to be accurate when dealing with imbalanced datasets) and our DMC. Our DMC generally achieves the lower maximum risks per class and better equalizes them than the others methods. This equalization is depicted in Fig. 3 on the DermaMinist database. The DBC and the initial CNN fail in this task.

CIFAR100 database. We also consider the CIFAR-100 database [38] which contains 60,000 images with $K = 100$ classes. We considered a training set, respectively a test set, composed of 40,000 images, resp. 20,000 images. Both the training and test sets satisfied the balanced class proportions

TABLE II
RESULTS ON THE VALIDATION AND TEST SETS FOR EACH OUTPUT CLASSIFIER FITTED ON THE EXTRACTED FEATURES.

	Classifier δ	DERMA Val	DERMA Test	BREAST Val	BREAST Test	OCT Val	OCT Test
ResNet-18							
$\hat{r}_\varphi(\delta)$	CNN	0.29	0.30	0.17	0.16	0.06	0.28
	DBC	0.26	0.32	0.14	0.18	0.07	0.20
	KNN	0.22	0.29	0.09	0.14	0.06	0.26
	SVM	0.00	0.33	0.00	0.20	0.02	0.28
	WRF	0.32	0.41	0.08	0.17	0.08	0.20
	DMC*	0.48	0.54	0.17	0.19	0.13	0.21
$\max_{k\in\mathcal{Y}} \hat{R}_k(\delta)$	CNN	1.00	1.00	0.43	0.50	0.35	0.76
	DBC	1.00	1.00	0.43	0.57	0.47	0.41
	KNN	1.00	1.00	0.19	0.21	0.41	0.73
	SVM	0.00	1.00	0.00	0.47	0.46	0.76
	WRF	0.43	0.91	0.12	0.24	0.27	0.52
	DMC*	0.54	0.83	0.19	0.19	0.13	0.32
$\psi(\delta)$	CNN	0.83	0.84	0.36	0.46	0.33	0.69
	DBC	0.90	0.87	0.39	0.54	0.46	0.33
	KNN	0.93	0.90	0.14	0.09	0.38	0.70
	SVM	0.00	0.83	0.00	0.20	0.44	0.73
	WRF	0.43	0.65	0.12	0.07	0.20	0.43
	DMC*	0.21	0.37	0.03	0.00	0.01	0.24
EfficientNet-B7							
$\hat{r}_\varphi(\delta)$	CNN	0.27	0.27	0.22	0.23	0.07	0.27
	DBC	0.24	0.28	0.19	0.24	0.08	0.25
	KNN	0.25	0.28	0.21	0.22	0.07	0.28
	SVM	0.25	0.27	0.22	0.22	0.07	0.28
	WRF	0.30	0.36	0.13	0.25	0.09	0.25
	DMC*	0.48	0.49	0.23	0.29	0.14	0.22
$\max_{k\in\mathcal{Y}} \hat{R}_k(\delta)$	CNN	1.00	0.87	0.48	0.59	0.41	0.72
	DBC	0.96	0.91	0.48	0.69	0.47	0.45
	KNN	1.00	1.00	0.38	0.52	0.41	0.73
	SVM	1.00	1.00	0.67	0.79	0.46	0.76
	WRF	0.83	0.84	0.14	0.43	0.27	0.52
	DMC*	0.43	0.83	0.24	0.52	0.15	0.29
$\psi(\delta)$	CNN	0.90	0.77	0.35	0.49	0.38	0.69
	DBC	0.82	0.87	0.39	0.61	0.44	0.36
	KNN	0.95	0.94	0.24	0.41	0.38	0.70
	SVM	0.96	0.96	0.61	0.78	0.44	0.73
	WRF	0.65	0.72	0.05	0.24	0.20	0.43
	DMC*	0.11	0.41	0.01	0.31	0.01	0.17

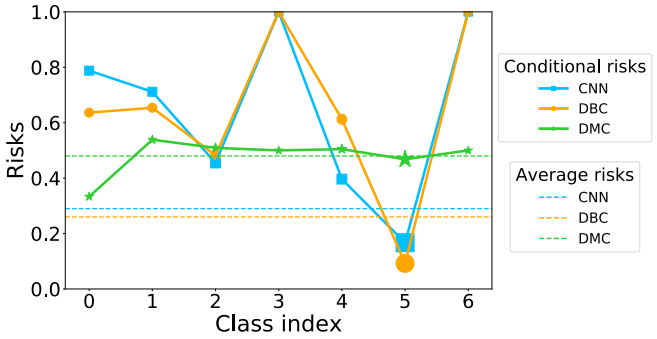

Fig. 3. Class-conditional risks from the DermaMNIST validation set for CNN ResNet-18. The size of each point depends on the class-proportions.

$\pi = [1/100, \ldots, 1/100]$. We considered the deep features extracted from the last hidden layer of the CNN EfficientNet-B0 [36], and we compared the DMC with the Weighted Logistic Regression (WLR) applied both on the deep features. Since the class proportions are perfectly balanced, it results that the weights of the WLR do not help. As illustrated

in Fig. 4, this classifier is not able to equalize the class-conditional risks. Despite these difficulties, we can observe that our approach minimizes significantly the maximum of the conditional risks on this large scale database.

Fig. 4. Class-conditional risks on the CIFAR-100 database.

IV. Conclusion

This paper presents a new approach to equalize the class-conditional classification probalities of a DNN. Our approach couples a trained DNN with a minimax classifier as the top layer. Future work will study the generalization error of this minimax classifier.

References

[1] G. Litjens *et al.*, "A survey on deep learning in medical image analysis," *Medical Image Analysis*, vol. 42, pp. 60–88, 2017.

[2] S. Korolev, A. Safiullin, M. Belyaev, and Y. Dodonova, "Residual and plain convolutional neural networks for 3d brain MRI classification," in *IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*, 2017. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/7950647

[3] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, 2018.

[4] M. A. Mazurowski, P. A. Habas, J. M. Zurada, J. Y. Lo, J. A. Baker, and G. D. Tourassi, "Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance," *Neural Networks*, vol. 21, pp. 427–436, 2008.

[5] J. Tian, Y.-C. Liu, N. Glaser, Y.-C. Hsu, and Z. Kira, "Posterior re-calibration for imbalanced datasets," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020.

[6] M. Kukar and I. Kononenko, "Cost-sensitive learning with neural networks," *European Conference on Artificial Intelligence*, 1998.

[7] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *Journal of Big Data*, vol. 6, no. 1, p. 27, Mar. 2019.

[8] K. Cao, C. Wei, A. Gaidon, N. Arechiga, and T. Ma, "Learning imbalanced datasets with label-distribution-aware margin loss," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, pp. 1567–1578.

[9] C. Gilet, S. Barbosa, and L. Fillatre, "Discrete box-constrained minimax classifier for uncertain and imbalanced class proportions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[10] D. Stutz, M. Hein, and B. Schiele, "Confidence-calibrated adversarial training: Generalizing to unseen attacks," in *Proceedings of the 37th International Conference on Machine Learning, ICML*, ser. Proceedings of Machine Learning Research, vol. 119. PMLR, 2020, pp. 9155–9166.

[11] L. He, Z. Guo, K. Huang, and Z. Xu, "Deep minimax probability machine," *2019 International Conference on Data Mining Workshops (ICDMW)*, pp. 869–876, 2019.

[12] M. Fauß, A. M. Zoubir, and H. V. Poor, "Minimax robust detection: Classic results and recent advances," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2252–2283, 2021.

[13] L. Fillatre, "Constrained epsilon-minimax test for simultaneous detection and classification," *IEEE Transactions on Information Theory*, vol. 57, no. 12, pp. 8055–8071, dec. 2011.

[14] L. Fillatre and I. Nikiforov, "Asymptotically uniformly minimax detection and isolation in network monitoring," *IEEE Transactions on Signal Processing*, vol. 60, no. 7, pp. 3357–3371, 2012.

[15] L. Fillatre, "Constructive minimax classification of discrete observations with arbitrary loss function," *Signal Processing*, vol. 141, pp. 322–330, 2017.

[16] A. Guerrero-Curieses, R. Alaíz-Rodríguez, and J. Cid-Sueiro, "A fixed-point algorithm to minimax learning with neural networks," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 34, pp. 383–392, 2004.

[17] J. Lee and M. Raginsky, "Minimax statistical learning with wasserstein distances," in *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds., vol. 31. Curran Associates, Inc., 2018.

[18] K. Bratanova, I. Kareev, and R. Salimov, "Minimax modifications of linear discriminant analysis for classification with rare classes," in *2020 IEEE East-West Design & Test Symposium (EWDTS)*, 2020, pp. 1–5.

[19] G. Lanckriet, L. Ghaoui, C. Bhattacharyya, and M. Jordan, "Minimax probability machine," in *Advances in Neural Information Processing Systems*, vol. 14. MIT Press, 2001.

[20] ——, "A robust minimax approach to classification," *Journal of Machine Learning Research*, vol. 3, 12 2002.

[21] K. Huang, H. Yang, I. King, M. R. Lyu, and L. Chan, "The minimum error minimax probability machine," *J. Mach. Learn. Res.*, vol. 5, p. 1253–1286, dec 2004.

[22] K. Huang, H. Yang, I. King, and M. R. Lyu, "Maximizing sensitivity in medical diagnosis using biased minimax probability machine," *IEEE Trans. Biomed. Eng.*, vol. 53, no. 5, pp. 821–831, 2006.

[23] F. Farnia and D. Tse, "A minimax approach to supervised learning," in *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[24] H. Kaizhu, Y. Haiqin, K. Irwin, R. L. Michael, and L. Chan, "The minimum error minimax probability machine," *Journal of Machine Learning Research*, pp. 1253–1286, 2004.

[25] J. Schmidt-Hieber, "Nonparametric regression using deep neural networks with ReLU activation function," *The Annals of Statistics*, vol. 48, no. 4, pp. 1875 – 1897, 2020.

[26] M. Kohler, A. Krzyżak, and S. Langer, "Estimation of a function of low local dimensionality by deep neural networks," *IEEE Transactions on Information Theory*, vol. 68, no. 6, pp. 4032–4042, 2022.

[27] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[28] N. Tajbakhsh, J. Y. Shin, S. R. Gurudu, R. T. Hurst, C. B. Kendall, M. B. Gotway, and J. Liang, "Convolutional neural networks for medical image analysis: Full training or fine tuning?" *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1299–1312, 2016.

[29] H. V. Poor, *An Introduction to Signal Detection and Estimation*, 2nd ed. Springer-Verlag New York, 1994.

[30] J. Dougherty, R. Kohavi, and M. Sahami, "Supervised and unsupervised discretization of continuous features," *International Conference on Machine Learning*, 1995.

[31] L. A. Dalton and E. R. Dougherty, "Bayesian minimum mean-square error estimation for classification error - part i: Definition and the bayesian mmse error estimator for discrete classification," *IEEE Transactions on Signal Processing*, vol. 59, pp. 115–129, 2011.

[32] Y. Yang and G. I. Webb, "Discretization for naive-bayes learning: managing discretization bias and variance," *Machine Learning*, vol. 74, no. 1, pp. 39–74, Jan 2009.

[33] L. Peng, W. Qing, and G. Yujia, "Study on comparison of discretization methods," *IEEE, International Conference on Artificial Intelligence and Computational Intelligence*, pp. 380–384, 2009.

[34] K. He, X. Zhang, S. Ren, and J. Sun, "*MedMNIST* databases," https://zenodo.org/record/4269852#.X_mdsulKiHE.

[35] ——, "Deep residual learning for image recognition," *CVPR*, 2016.

[36] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International Conference on Machine Learning*, 2019, pp. 6105–6114.

[37] J. Yang, R. Shi, and B. Ni, "Medmnist classification decathlon: A lightweight automl benchmark for medical image analysis," *arXiv:2010.14925*, 2020.

[38] A. Krizhevsky, "Learning multiple layers of features from tiny images," 2009, https://www.cs.toronto.edu/ kriz/cifar.html.