Softmin Discrete Minimax Classifier for Imbalanced Classes and Prior Probability Shifts

Cyprien Gilet, Marie Guyomard, Sébastien Destercke, Lionel Fillatre

SOFTMIN DISCRETE MINIMAX CLASSIFIER FOR IMBALANCED CLASSES AND PRIOR PROBABILITY SHIFTS

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

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

Sébastien Destercke
Université de Technologie de Compiègne
CNRS, Heudiasyc Laboratory, France
sebastien.destercke@hds.utc.fr

Lionel Fillatre
Université Côte d'Azur
CNRS, I3S Laboratory, France
lionel.fillatre@i3s.unice.fr

ABSTRACT

This paper proposes a new approach for dealing with imbalanced classes and prior probability shifts in supervised classification tasks. Coupled with any feature space partitioning method, our criterion aims to compute an almost-Bayesian randomized equalizer classifier for which the maxima of the class-conditional risks are minimized. Our approach belongs to the historically well-studied field of randomized minimax criteria. Our new criterion can be considered as a self-sufficient classifier, or can be easily coupled with any pretrained Convolutional Neural Networks and Decision Trees to address the issues of imbalanced classes and prior probability shifts. Numerical experiments compare our criterion to several state-of-the-art algorithms and show the relevance of our approach when it is necessary to well classify the minority classes and to equalize the risks per class. Experiments on the CIFAR-100 database show that our criterion scales well when the number of classes is large.

Keywords Imbalanced classes · Prior probability shifts · Minimax classifier · Randomized Decision rule

1 Introduction

Supervised classification is becoming increasingly used in several real applications such as precision medicine, condition monitoring or fraud detection. Given $K \geq 2$ the number of classes, the objective is to predict the true class of samples in the set of labels $\mathcal{Y} = \{1, \ldots, K\}$ from the attributes (also called features) describing each instance.

As historically studied in statistical decision theory (Ferguson, 1967; Berger, 1985; Poor, 1994), both randomized or non-randomized (also called deterministic) decision rules can be relevant for these classification tasks. While deterministic classifiers always assign the same class $k \in \mathcal{Y}$ to a given fixed feature profile, randomized classifiers will assign the class $k \in \mathcal{Y}$ at random with some estimated probability from this given fixed feature profile. In the following, denoting $\mathcal{X}$ the feature space of all attribute values, let $\Delta := \{\delta : \mathcal{X} \to \mathcal{Y}\}$ be the set of all possible classifiers and let $\mathcal{D} \subset \Delta$ denote the set of deterministic classifiers only.

The issues of imbalanced class proportions and prior probability shifts have been actively studied in statistical decision theory for the past century (Ferguson, 1967; Berger, 1985; Poor, 1994) and remain important to solve for supervised machine learning classifiers (He and Garcia, 2009; Buda et al., 2018; Gilet et al., 2020; Cao et al., 2019; Tian et al., 2020; Quiñonero-Candela et al., 2008).

1.1 Empirical average risk minimization

Given a multiset $\mathcal{S} = \left\{(Y_i, X_i), i \in \mathcal{I}\right\}$ of m labeled training samples, the usual objective in learning an accurate classifier (Vapnik, 1999; Hastie et al., 2009) is generally to train a decision rule $\delta \in \Delta$ which assigns to each sample

$i \in \mathcal{I}$ a class $\delta(X_i) = Y_i \in \mathcal{Y}$ from its feature vector $X_i \in \mathcal{X}$, and such that δ minimizes the empirical average risk of classification errors

$$\hat{r}(\delta) = \frac{1}{m} \sum_{i \in \mathcal{I}} L(Y_i, \delta(X_i)). \tag{1}$$

In equation (1), $L : \mathcal{Y} \times \mathcal{Y} \to [0, +\infty)$ denotes the loss function such that, for all $(k, l) \in \mathcal{Y} \times \mathcal{Y}$, $L(k, l) := L_{kl}$ corresponds to the loss, or the cost, of predicting the class l whereas the real class is k.

1.2 Dealing with imbalanced classes

Let $\hat{\pi} := [\hat{\pi}_1, \ldots, \hat{\pi}_K]$ denote the class proportions of the training set, such that for all class $k \in \mathcal{Y}$, $\hat{\pi}_k := \frac{1}{m} \sum_{i \in \mathcal{I}} \mathbb{1}_{\{Y_i = k\}}$. As explained in (Ferguson, 1967; Berger, 1985; Poor, 1994; Gilet et al., 2020), the average risk of classification errors (1) associated with any classifier $\delta \in \Delta$ can be written as

$$\hat{r}\left(\delta\right) = \sum_{k \in \mathcal{Y}} \hat{\pi}_k \hat{R}_k\left(\delta\right), \tag{2}$$

where for $k \in \mathcal{Y}$, $\hat{R}_k\left(\delta\right)$ is the empirical class-conditional risk associated with class k, defined by

$$\hat{R}_k\left(\delta\right) := \sum_{l \in \mathcal{Y}} L_{kl}\, \hat{\mathbb{P}}(\delta(X_i) = l \mid Y_i = k). \tag{3}$$

In equation (3), $\hat{\mathbb{P}}(\delta(X_i) = l \mid Y_i = k)$ characterizes the empirical probability for the decision rule δ to predict the class l given that the true class is k.

It is well known in the literature (He and Garcia, 2009; Japkowicz and Stephen, 2002; Chawla et al., 2002; Elkan, 2001; Dong et al., 2019; Gilet et al., 2020; Xu et al., 2020) that when the class proportions $\hat{\pi}$ are imbalanced, and as a consequence of (2), learning a classifier by minimizing (1) generally leads the minority classes to have a large conditional risk. As underlined in (Cui et al., 2019; Buda et al., 2018; Mazurowski et al., 2008; Tian et al., 2020), this issue also occurs to Convolutional Neural Networks classifiers, like for instance in precision medicine (Litjens et al., 2017; Colliot and Burgos, 2020).

A common approach to deal with imbalanced datasets is to balance the data by resampling the training set when the number of samples is large enough (Japkowicz and Stephen, 2002; He and Garcia, 2009). However, this approach introduces a bias since the actual state of nature can remain imbalanced. Another common approach is cost-sensitive learning, studied in (Ávila Pires et al., 2013; Drummond and C. Holte, 2003; Japkowicz and Stephen, 2002; He and Garcia, 2009; Kukar and Kononenko, 1998), which aims to optimize the cost of class classification errors in order to counterbalance the number of occurrences of each class. However, these costs are generally difficult to optimize when dealing with a large number of classes (Gilet et al., 2020). Buda et al. (2018) provide an interesting overview of approaches to address the issue of imbalanced datasets in deep learning. For instance, *thresholding* (Lawrence et al., 1998), *one-class classification* (Lee and Cho, 2006), *hybrid of methods* (Chawla et al., 2003), also attempt to address this issue of imbalanced classes. More recently, Cao et al. (2019) suggest to replace the standard cross-entropy objective during the training procedure. Tian et al. (2020) propose a post-training prior rebalancing method.

While many approaches have been studied to address this issue of imbalanced classes, statistical decision theory in (Ferguson, 1967; Berger, 1985; Poor, 1994) showed that the optimal criterion to minimize the maximum of the class-conditional risks is the *Minimax Classifier* (see Subsection 1.4). This is indeed the Bayesian classifier for which the risks per class are all minimized and balanced. However, learning a minimax classifier is difficult in Machine Learning, especially when dealing with several classes and any kind of loss function L. Nowadays only a few minimax algorithms have been proposed to deal with these general contexts (Guerrero-Curieses et al., 2004; Gilet et al., 2020), which is the scope of our present paper.

1.3 Dealing with prior probability shifts

Prior probability shift (Moreno-Torres et al., 2012; Quiñonero-Candela et al., 2008) characterizes an evolution in the distribution of the priors between the training set and test samples. A concrete example of prior shifts can occur when diagnosing the flu: the proportion of sick patients is not the same in October, January or July. More generally, prior shifts can occur in several real application fields and are usually caused by unknown attributes. It is therefore not possible to predict when this issue can occur. Nowadays, the issue of prior probability shifts is more and more discussed in the Machine Learning field (Gilet et al., 2020; Tian et al., 2020; Quiñonero-Candela et al., 2008; Moreno-Torres et al., 2012) and remains essential to solve in many real application domains.

Indeed, the damage caused by prior shifts are the following: Denoting,

$$\mathbb{S} := \left\{ \pi \in [0,1]^K : \sum_{k \in \mathcal{Y}} \pi_k = 1 \right\} \tag{4}$$

the K-dimensional simplex, Poor (1994); Berger (1985); Gilet et al. (2020); Ferguson (1967) explain that the average risk of classification error associated with any fitted classifier $\delta \in \Delta$ and as a function of any prior shift $\pi \in \mathbb{S}$ is equal to

$$\hat{r}(\pi, \delta) = \sum_{k \in \mathcal{Y}} \pi_k \hat{R}_k(\delta). \tag{5}$$

Since the $\hat{R}_k(\delta)$'s do not depend on π, the risk (5) is a linear function with respect to π and can dramatically increase until $\max_{k \in \mathcal{Y}} \hat{R}_k(\delta)$ when predicting test samples. An illustration of this phenomenon is provided in Appendix A.

As discussed in (Gilet et al., 2020; Tian et al., 2020; González et al., 2017), the sensitivity of a classifier to prior probability shifts is therefore greater when the class-conditional risks (3) are imbalanced. Hence, one relevant approach to address this issue is to minimize $\max_{k \in \mathcal{Y}} \hat{R}_k(\delta)$ during the training step, and thus to balance all the risks per class (3), which is the scope of the *Minimax Criterion* (see Subsection 1.4). The average risk (5) would thus remain constant and stable whatever the shift in the priors. An illustration of this desire robustness is illustrated in Fig. 2, Right. This objective is also the scope in our present paper.

1.4 Minimax criterion

Prior probability shifts and training with imbalanced datasets share therefore a common trait, namely the sensitivity to unequal class-conditional risks. Equalizing the class-conditional risks (3) is thus essential to obtain a robust classifier face to these prior issues. As emphasized in (Ferguson, 1967; Yablon and Chu, 1982; Berger, 1985; Poor, 1994; Duda et al., 2000; Guerrero-Curieses et al., 2004; Gilet et al., 2020), a famous and relevant approach to address these issues is to fit a minimax classifier. A minimax classifier seeks to minimize the maximum of the class-conditional risks during the training step. Hence, this approach tends to equalize these risks per class and makes the average risk of error (5) robust to any prior probability shift.

In statistical decision theory (Ferguson, 1967; Berger, 1985; Poor, 1994), a minimax classifier is usually fitted by maximizing the Bayes risk with respect to the prior probabilities over the simplex $\mathbb{S}$. However, learning a minimax classifier is difficult in Machine Learning, especially when dealing with several classes and any kind of loss function L (Gilet et al., 2020; Guerrero-Curieses et al., 2004). Indeed in most real application fields the calculation of the empirical Bayes risk over the simplex is generally intractable because of the curse of dimensionality. Furthermore, in many real application fields we often have to deal with both numeric and categorical features, many of them presenting dependencies. Hence, computing a good estimate of the feature joint distribution in each class in order to achieve a good estimate of the empirical Bayes risk over the simplex $\mathbb{S}$ remains highly complicated.

For all these reasons, we proposed in our previous research (Gilet et al., 2020; Gilet et al., 2019) to beforehand partition the feature space and then to learn the minimax classifier by using a closed-form expression of the Bayes risk over the simplex $\mathbb{S}$. We showed that the discrete empirical Bayes risk is a concave non-differentiable multivariate piecewise affine function with respect to the priors. This Discrete Minimax Classifier is deterministic and corresponds to the discrete Bayes classifier associated with the priors that maximize this multivariate piecewise affine function. This approach can outperform several other state of the art methods for minimizing the maximum of the class-conditional risks (3). However, depending on the feature space partitioning, this criterion does not always guarantee an optimal equalization of these risks per class due to the non-differentiability of the discrete Bayes risk.

1.5 Contributions and organization of the paper

The objective of this paper is to design a new criterion which aims to minimize and equalize all the class-conditional risks (3) in the context of beforehand partitioned feature space, as in (Gilet et al., 2020; Gilet et al., 2019, 2020). Our new criterion can be considered as a self-sufficient classifier or can be easily coupled with any pretrained Convolutional Neural Networks or Decision Trees for dealing with the previously emphasized prior issues. Our contributions are the following:

- Section 2 argues the interests of considering our novel criterion. We show that if the risks per class of the deterministic discrete minimax criterion (Gilet et al., 2020) are not all equalized, there exists an almost-Bayesian randomized classifier which can achieve a lower maximum class-conditional risk and which can equalize all the risks per class. To this aim, we consider randomized decision rules which are relevant to achieve this scope, as emphasized in (Ferguson, 1967; Berger, 1985; Poor, 1994). Moreover, another major benefit of our new classifier is to provide estimated probability scores of prediction for each class.

- Section 3 explains how to compute our new criterion called *Softmin Discrete Minimax Classifier*. We first analytically approximate the discrete empirical Bayes risk over the simplex $\mathbb{S}$ using a softmin randomized decision rule which converges to the discrete Bayes classifier depending on a temperature parameter $\lambda > 0$. Secondly, we compute the priors $\pi^\star \in \mathbb{S}$ which allow to equalize all the class-conditional risks (3). To this aim, we show that these priors $\pi^\star$ are a root of a specific non-convex application over the simplex $\mathbb{S}$. Finally, our resulting classifier is a softmin randomized decision rule with respect to the priors $\pi^\star$.
- Section 4 compares our approach to several state-of-the-art algorithms on six real databases and show the relevance of our criterion when it is necessary to well classify the minority classes and to equalize all the class-conditional risks. Experiments on the CIFAR-100 database show that our criterion is scalable when the number of classes is large.
- Section 5 concludes the paper. We discuss the asset of our new criterion to provide probability scores of prediction in each class, which could open an interesting path to minimax learning for multi-labels decision-making.

2 Interest of our new criterion

This section reminds the principle of the deterministic discrete minimax criterion established in (Gilet et al., 2020) and presents our new objectives in order to improve classification performances.

2.1 Partitioning then classifying

In order to well approximate the Bayes risk when processing numeric or mixed features in a high dimensional feature space, a relevant approach is to partition the feature space which allows to analytically calculate the discrete Bayes risk (Devroye et al., 1996; Braga-Neto and Dougherty, 2005; Dalton and Dougherty, 2011; Gilet et al., 2020). In other words, the feature space $\mathcal{X}$ is partitioned into T disjoint regions $\{\Omega_1, \ldots, \Omega_T\}$ such that $\cup_{t=1}^{T} \Omega_t = \mathcal{X}$. This defines a mapping $\phi : \mathcal{X} \mapsto \mathcal{T} := \{1, \ldots, T\}$ such that $\phi(X_i) = t$ if and only if $X_i \in \Omega_t$. The reduced dimension T is chosen to get a good trade-off between accuracy and scalability of the generalization error.

For instance, the discrete profiles $t \in \mathcal{T}$ can correspond to the leaves of a tree partitioning (as illustrated in Figure 1, left) when considering a partitioning using supervised decision trees (Breiman et al., 1984; Scott and Nowak, 2006; Gilet et al., 2020). They can also correspond to the centroids of each Voronoi cell after a Kmeans partitioning (MacQueen, 1967). The Kmeans partitioning considers that the instances belonging to the same Voronoi cell Ω_t have similar behavior and may belong to the same class. This philosophy is closely related to clustering of bandits based approaches for which the objective is to identify clusters of users so that the users belonging to the same cluster are supposed to have similar behavior, which allows to improve the contents recommendation based on the payoffs computed in each cluster (Gentile et al., 2014; Li et al., 2016).

2.2 Reminds on the deterministic discrete minimax criterion

Let $\mathcal{I}_k = \{i \in \mathcal{I} : Y_i = k\}$ denote the set of learning instances from class $k \in \mathcal{Y}$ and $m_k = |\mathcal{I}_k|$ the number of instances in $\mathcal{I}_k$. Dealing with discrete profiles $\mathcal{T} = \phi(\mathcal{X})$, we can estimate from the training set the probabilities $\hat{p}_{kt}$ of observing the discrete profile t given that the class label is k, for all $t \in \mathcal{T}$ and for all $k \in \mathcal{Y}$, such that

$$\hat{p}_{kt} := \frac{1}{m_k} \sum_{i \in \mathcal{I}_k} \mathbb{1}_{\{\phi(X_i)=t\}}. \tag{6}$$

From equation (6), we calculated analytically in (Gilet et al., 2020) the deterministic discrete Bayes classifier $\delta_\pi^B := \operatorname{argmin}_{\delta \in \mathcal{D}} \hat{r}(\pi, \delta)$ associated with any prior $\pi \in \mathbb{S}$. This deterministic discrete Bayes classifier is given by

$$\delta_\pi^B : X_i \mapsto \operatorname*{argmin}_{l \in \mathcal{Y}} f_l(\pi, X_i), \tag{7}$$

with for each class $l \in \mathcal{Y}$,

$$f_l(\pi, X_i) := \sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{Y}} L_{kl}\, \pi_k\, \hat{p}_{kt}\, \mathbb{1}_{\{\phi(X_i)=t\}}. \tag{8}$$

The classifier (7) is deterministic in the sense that a unique class $k \in \mathcal{Y}$ is assigned to each discrete profile $t = \phi(X_i) \in \mathcal{T}$. This firmness also occurs when $\operatorname{argmin}_{l \in \mathcal{Y}} f_l(\pi, X_i)$ is not unique. In this special case, the deterministic classifier (7) arbitrarily and constantly selects the same class label $l \in \operatorname{argmin}_{l \in \mathcal{Y}} f_l(\pi, X_i)$.

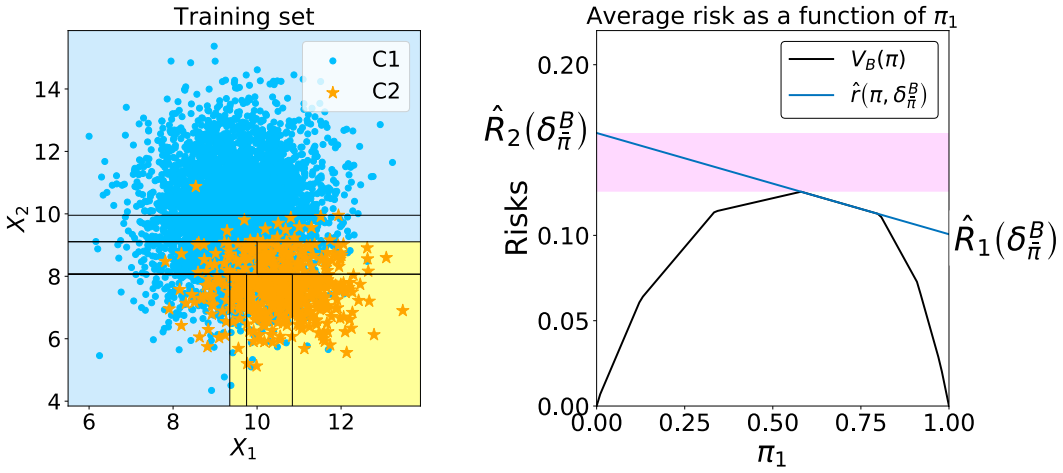

Figure 1: Experiments on a synthetic dataset with $K = 2$ classes and $d = 2$ numeric features ($\mathcal{X} \subset \mathbb{R}^2$) generated using Scikit-Learn (Pedregosa et al., 2011). **Left.** Decision boundaries of the discrete minimax classifier (Gilet et al., 2020) applied on the tree partitioning $\phi : \mathcal{X} \to \mathcal{T}$ which designed $T = 8$ discrete profiles (the number of leaves). **Right.** The surface V_B corresponds to the empirical Bayes risk (9) on the tree partitioning as the function of the priors $\pi \in \mathbb{S}$. Moreover, $\hat{r}\left(\pi, \delta^B_{\bar{\pi}}\right)$ characterizes the risk of the discrete minimax classifier face to prior probability shifts (5). Since we have $K = 2$ classes, this risk can be written as a linear function of π_1: $\hat{r}\left(\pi, \delta^B_{\bar{\pi}}\right) = \pi_1[\hat{R}_1\left(\delta^B_{\bar{\pi}}\right) - \hat{R}_2\left(\delta^B_{\bar{\pi}}\right)] + \hat{R}_2\left(\delta^B_{\bar{\pi}}\right)$.

Furthermore, in (Gilet et al., 2020), we calculated analytically the average risk (1) of the discrete Bayes classifier δ^B_{π} as a function of the priors $\pi \in \mathbb{S}$. This discrete empirical Bayes risk is given by

$$V_B : \pi \mapsto \min_{\delta \in \mathcal{D}} \hat{r}(\pi, \delta) = \hat{r}\left(\delta^B_{\pi}\right) = \sum_{k \in \mathcal{Y}} \pi_k \hat{R}_k\left(\delta^B_{\pi}\right), \tag{9}$$

where for all $k \in \mathcal{Y}$,

$$\hat{R}_k\left(\delta^B_{\pi}\right) = \sum_{t \in \mathcal{T}} \sum_{l \in \mathcal{Y}} L_{kl}\, \hat{p}_{kt}\, \mathbb{1}_{\{\xi_{lt} = \min_{q \in \mathcal{Y}} \xi_{qt}\}}, \tag{10}$$

with, for all $l \in \mathcal{Y}$ and all $t \in \mathcal{T}$, $\xi_{lt} = \sum_{k \in \mathcal{Y}} L_{kl}\, \pi_k\, \hat{p}_{kt}$. We showed that V_B is a non-differentiable concave multivariate piecewise affine function over the simplex $\mathbb{S}$ (as illustrated in Figure 1, right). The non-differentiability of V_B occurs in the finite set of priors $\Pi^c = \mathbb{S} \backslash \Pi$, where $\Pi \subset \mathbb{S}$ denotes the set of priors for which $\operatorname{argmin}_{l \in \mathcal{Y}} f_l(\pi, X_i)$ is unique.

Finally, the discrete minimax classifier (Gilet et al., 2020), denoted by $\delta^B_{\bar{\pi}}$, is the discrete Bayes classifier (7) associated with the priors $\bar{\pi} = \operatorname{argmax}_{\pi \in \mathbb{S}} V_B(\pi)$. It corresponds to the non-randomized Bayesian classifier for which the class-conditional risks are the most balanced on the partitioned feature space.

2.3 Can we perform better?

In (Gilet et al., 2020), we showed that the deterministic discrete minimax criterion $\delta^B_{\bar{\pi}}$ can outperform several other state of the art Machine Learning methods for minimizing the maximum of the class-conditional risks (3). However, since the empirical Bayes risk V_B defined in equation (9) is generally not differentiable at the least favorable priors $\bar{\pi} = \operatorname{argmax}_{\pi \in \mathbb{S}} V_B(\pi)$, $\delta^B_{\bar{\pi}}$ is not necessarily an equalizer classifier. Sometimes, this criterion may appear unable to balance the class-conditional risks (3), even though it reaches the lowest maximum risk per class in the non-randomized Bayesian sense. Figure 1, right, illustrates this phenomenon for $K = 2$ classes.

A decision rule $\delta \in \Delta$ is called an equalizer classifier if its class-conditional risks (3) are all equal, that is $\hat{R}_1(\delta) = \cdots = \hat{R}_K(\delta)$. Statistical decision theory in (Ferguson, 1967; Berger, 1985; Poor, 1994; Borovkov, 1998) showed that an equalizer Bayesian classifier is necessarily a minimax classifier, but if a non-randomized minimax classifier is not equalizer, randomizing it is therefore a relevant solution. For instance in Figure 1, right, we can observe that any randomized equalizer classifier $\delta^\star \in \Delta$, for which the average risk (5) shifts only on the pink area, becomes more robust since such a classifier would achieve $\max_{k \in \mathcal{Y}} \hat{R}_k(\delta^\star) \leq \max_{k \in \mathcal{Y}} \hat{R}_k(\delta^B_{\bar{\pi}})$.

2.4 Scope of the present paper

The scope of this paper is therefore to compute such a randomized equalizer classifier $\delta^\star \in \Delta$ when the deterministic discrete minimax classifier $\delta_\pi^B \in \mathcal{D} \subset \Delta$ is not equalizer. In this paper, we consider the same following assumptions as for the discrete minimax criterion (Gilet et al., 2020). These assumptions are not restrictive in practice but necessary to the theoretical development of our approach.

Assumption 1. *Since we can only exploit the instances from the training set, the probabilities $\hat{p}_{kt}$ defined in (6) are assumed to be estimated once and for all. This is a usual assumption in the literature (González et al., 2017; Gilet et al., 2020). By estimating these probabilities using the full training set as in (6), we get the best unbiased estimate with the smallest variance (Rao, 1973). We then consider these probabilities $\hat{p}_{kt}$ fixed.*

Assumption 2. *The dataset is sensitive to imbalanced priors in the common sense that, for all class $k \in \mathcal{Y}$ there exists $\varepsilon_k > 0$ such that for all $\pi \in \mathcal{Q}_k := \{\pi \in \mathbb{S} : \pi_k < \varepsilon_k\}$, $\hat{R}_k(\delta_\pi) \geq \hat{r}(\delta_\pi)$, where $\delta_\pi \in \Delta$ is a classifier fitted when considering the priors $\pi \in \mathbb{S}$. Moreover, for each class $k \in \mathcal{Y}$, there exists $\eta_k > 0$ such that for all $\pi \in \mathcal{U}_k := \{\pi \in \mathbb{S} : \pi_k > 1 - \eta_k\}$, $\hat{R}_k(\delta_\pi) \leq \hat{r}(\delta_\pi)$.*

3 Softmin Discrete Minimax Criterion

In this section we design our novel randomized criterion for Machine Learning tasks which aims to perform better than the deterministic discrete minimax classifier for minimizing the maximum of the class-conditional risks on the same partitioned feature space.

3.1 Softmin randomized decision rule

We wish to approximate the deterministic discrete Bayes classifier (7) by an almost-Bayesian randomized decision rule $\delta^\star \in \Delta$ which assigns a label $k \in \mathcal{Y}$ with probability $\mathbb{P}\left(\delta^\star(X_i) = k\right)$. In order to well approximate the Bayes classifier (7) with respect to any priors $\pi \in \mathbb{S}$, we therefore wish to assign the label k to an instance X_i with high probability if $k = \operatorname{argmin}_{l \in \mathcal{Y}} f_l(\pi, X_i)$. This goal naturally leads us to consider the softmin criterion, especially since the softmax (similar to softmin) decision rule is actively used as output layer of Convolutional Neural Networks (Goodfellow et al., 2016). The following definition presents the almost-Bayesian randomized softmin classifier.

Definition 1. *Given a temperature parameter $\lambda > 0$ and when considering the quantities $f_1(\pi, X_i), \ldots, f_K(\pi, X_i)$, defined in equation (8), the softmin randomized discrete classifier δ_π^λ associated with the priors $\pi \in \mathbb{S}$ assigns the class label $k \in \mathcal{Y}$ at random with probability*

$$\hat{\mathbb{P}}\left(\delta_\pi^\lambda(X_i) = k\right) \;=\; \frac{e^{-\lambda\, f_k(\pi, X_i)}}{\sum_{l=1}^K e^{-\lambda\, f_l(\pi, X_i)}}. \tag{11}$$

This randomized classifier δ_π^λ depends on the temperature parameter $\lambda > 0$ and on the priors $\pi \in \mathbb{S}$. The following proposition estimates analytically the risk of this randomized softmin classifier over $\mathbb{S}$.

Proposition 1. *For any fixed temperature parameter $\lambda > 0$, the average risk (2) of the softmin discrete classifier δ_π^λ associated with any priors $\pi \in \mathbb{S}$ is given by $V_\lambda : \mathbb{S} \to [0, +\infty)$ such that*

$$V_\lambda(\pi) \;:=\; \hat{r}\left(\delta_\pi^\lambda\right) \;=\; \sum_{k \in \mathcal{Y}} \pi_k\, \hat{R}_k\left(\delta_\pi^\lambda\right), \tag{12}$$

where for all $k \in \mathcal{Y}$, the class-conditional risks $\hat{R}_k\left(\delta_\pi^\lambda\right)$ are analytically given by

$$\hat{R}_k\left(\delta_\pi^\lambda\right) = \sum_{t \in \mathcal{T}} \sum_{l \in \mathcal{Y}} L_{kl}\, \hat{p}_{kt}\, \frac{e^{-\lambda \sum_{j \in \mathcal{Y}} L_{jl}\, \pi_j\, \hat{p}_{jt}}}{\sum_{q=1}^K e^{-\lambda \sum_{j \in \mathcal{Y}} L_{jq}\, \pi_j\, \hat{p}_{jt}}}. \tag{13}$$

The proof of Proposition 1 is detailed in Appendix B.1. Similarly to the softmin discrete classifier δ_π^λ, its average risk $V_\lambda(\pi)$ also depends on the temperature parameter $\lambda > 0$. Since we wish our classifier to be equalizer and to achieve average risks lower than $\max_{k \in \mathcal{Y}} \hat{R}_k\left(\delta_\pi^B\right)$ (for instance to be within the pink area in Figure 1, right), the temperature parameter λ will play an important role. The following proposition studies the behavior of δ_π^λ and its average risk $V_\lambda(\pi)$ with respect to this parameter λ.

Proposition 2. *The randomized softmin classifier δ_π^λ associated with the priors $\pi \in \Pi$ (where $\Pi \subset \mathbb{S}$ is the set of priors for which $\operatorname{argmin}_{l \in \mathcal{Y}} f_l(\pi, X_i)$ is unique), converges in probability to the deterministic Bayes classifier δ_π^B (7) as λ goes to infinity. Furthermore, for any fixed priors $\pi \in \mathbb{S}$, the average risk $V_\lambda(\pi)$ converges pointwise to the Bayes risk $V_B(\pi)$ defined in (9) as λ goes to infinity.*

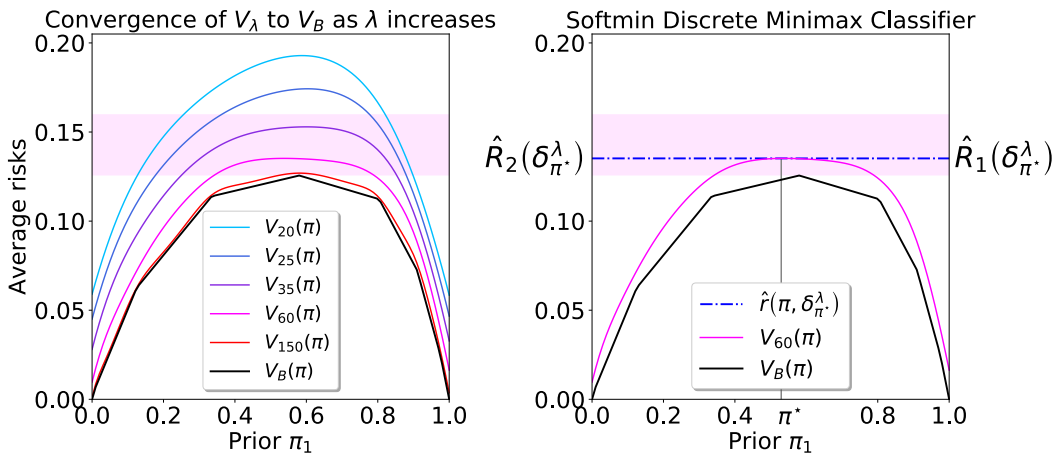

Figure 2: Experiments on the synthetic dataset introduced in Figure 1. The pink area corresponds to the set of average risks bounded by $V_B(\bar{\pi})$ and $\max_{k\in\mathcal{Y}} \hat{R}_k\left(\delta^B_{\bar{\pi}}\right)$. **Left.** Illustration of Proposition 2 and Corollary 1. As λ increases, $V_\lambda(\pi)$ converges pointwise to the Bayes risk $V_B(\pi)$. **Right.** For $\lambda = 60 \in \Lambda(\bar{\pi})$, the priors $\pi^\star \in \mathcal{B}_\lambda(\bar{\pi})$ allow the randomized classifier $\delta^\lambda_{\pi^\star}$ to be equalizer and to achieve lower average risks than $\max_{k\in\mathcal{Y}} \hat{R}_k\left(\delta^B_{\bar{\pi}}\right)$. Since we have $K = 2$ classes, the average risk of $\delta^\lambda_{\pi^\star}$ can be written as a linear function of π_1: $\hat{r}\left(\pi, \delta^\lambda_{\pi^\star}\right) = \pi_1[\hat{R}_1\left(\delta^\lambda_{\pi^\star}\right) - \hat{R}_2\left(\delta^\lambda_{\pi^\star}\right)] + \hat{R}_2\left(\delta^\lambda_{\pi^\star}\right)$.

The proof of Proposition 2 is detailed in Appendix B.2. While the deterministic discrete Bayesian classifier (7) assigns a unique class $k \in \mathcal{Y}$ to each discrete profile $t \in \mathcal{T}$, the softmin randomized decision rule δ^λ_π relaxes this firmness based on the probabilistic decision rule (11). According to Proposition 2, the more λ increases, the firmer δ^λ_π becomes.

Given $\bar{\pi} = \operatorname{argmax}_{\pi\in\mathbb{S}} V_B(\pi)$, if the deterministic discrete minimax criterion $\delta^B_{\bar{\pi}}$ is not equalizer, let us remind that we aim to reach average risks between $V_B(\bar{\pi})$ and $\max_{k\in\mathcal{Y}} \hat{R}_k\left(\delta^B_{\bar{\pi}}\right)$, e.g., within the pink area in Figure 1, right. Let us define the set of temperature parameters satisfying this goal as

$$\Lambda(\bar{\pi}) = \left\{\lambda > 0 : V_B(\bar{\pi}) \leq \max_{\pi\in\mathbb{S}} V_\lambda(\pi) \leq \max_{k\in\mathcal{Y}} \hat{R}_k\left(\delta^B_{\bar{\pi}}\right)\right\} \tag{14}$$

and then, for each acceptable temperature $\lambda \in \Lambda(\bar{\pi})$, we moreover define the set of acceptable priors

$$\mathcal{B}_\lambda(\bar{\pi}) = \left\{\pi \in \mathbb{S} : V_B(\bar{\pi}) \leq V_\lambda(\pi) \leq \max_{k\in\mathcal{Y}} \hat{R}_k\left(\delta^B_{\bar{\pi}}\right)\right\}. \tag{15}$$

Corollary 1. *Given $\bar{\pi} = \operatorname{argmax}_{\pi\in\mathbb{S}} V_B(\pi)$, if the deterministic discrete minimax criterion $\delta^B_{\bar{\pi}}$ is not equalizer, then the sets $\Lambda(\bar{\pi})$ and $\mathcal{B}_\lambda(\bar{\pi})$, $\lambda \in \Lambda(\bar{\pi})$, are not empty.*

The proof of Corollary 1 is detailed in Appendix B.3. Figure 2, left, illustrates Proposition 2 and Corollary 1 as λ increases when considering the toy example introduced in Figure 1. We can observe that for all acceptable temperature parameter $\lambda \in \Lambda(\bar{\pi})$ and for all associated priors $\pi \in \mathcal{B}_\lambda(\bar{\pi})$, $V_\lambda(\pi)$ belongs to the pink area, which is one of our objectives regarding the average risk of our classifier. The scope of the following subsection is now to demonstrate that given an acceptable temperature parameter $\lambda \in \Lambda(\bar{\pi})$ we can find some priors $\pi^\star \in \mathcal{B}_\lambda(\bar{\pi})$ such that the randomized decision rule $\delta^\lambda_{\pi^\star}$ is equalizer, like for instance in Figure 2, right.

3.2 Computation of the Softmin-DMC

We now assume that the deterministic discrete minimax decision rule $\delta^B_{\bar{\pi}}$ is not equalizer. We moreover consider an acceptable fixed temperature parameter $\lambda \in \Lambda(\bar{\pi})$ and we wish to find $\pi^\star \in \mathcal{B}_\lambda(\bar{\pi})$ such that the randomized decision rule $\delta^\lambda_{\pi^\star}$ is equalizer, like for instance in Figure 2, right.

For the following, let us consider the application $G : \mathbb{S} \to \mathbb{R}^K$ defined by

$$G(\pi) := \begin{bmatrix} g_1(\pi) \\ \vdots \\ g_K(\pi) \end{bmatrix}, \tag{16}$$

where for each class $k \in \mathcal{Y}$

$$g_k(\pi) := \hat{R}_k\left(\delta_\pi^\lambda\right) - V_\lambda(\pi). \tag{17}$$

The application $G(\pi)$ measures the gap between the average risk $V_\lambda(\pi)$ and each class-conditional risk $\hat{R}_k\left(\delta_\pi^\lambda\right)$, and can be analytically calculated for all priors $\pi \in \mathbb{S}$ from equations (12) and (13). The following lemma provides a necessary and sufficient condition ensuring that a softmin decision rule $\delta_{\pi^\star}^\lambda$ is equalizer.

Lemma 1. *For any fixed temperature parameter $\lambda \in \Lambda(\bar{\pi})$, the softmin randomized decision rule $\delta_{\pi^\star}^\lambda$ associated with the priors $\pi^\star \in \mathcal{B}_\lambda(\bar{\pi})$ is equalizer if and only if $G(\pi^\star) = 0$. Moreover, for any fixed parameter $\lambda \in \Lambda(\bar{\pi})$, such a root $\pi^\star \in \mathcal{B}_\lambda(\bar{\pi})$ exists.*

Theorem 1. *If the deterministic discrete minimax classifier $\delta_{\bar{\pi}}^B \in \mathcal{D}$ is not equalizer on a partitioned feature space $\mathcal{T} = \phi(\mathcal{X})$, then for all $\lambda \in \Lambda(\bar{\pi})$, the almost-Bayesian randomized equalizer decision rule $\delta_{\pi^\star}^\lambda$ associated with the priors $\pi^\star \in \mathcal{B}_\lambda(\bar{\pi})$ satisfying $G(\pi^\star) = 0$, allows to achieve, on this same partitioned feature space $\mathcal{T}$: $\max_{k\in\mathcal{Y}} \hat{R}_k(\delta_{\pi^\star}^\lambda) \leq \max_{k\in\mathcal{Y}} \hat{R}_k(\delta_{\bar{\pi}}^B)$.*

The proof of Lemma 1 is detailed in Appendix B.4 and the proof of Theorem 1 is a direct consequence of Lemma 1 and Corollary 1. While the deterministic discrete minimax criterion aims to minimize the maximum of the class-conditional risks on a partitioned feature space, the main asset of Theorem 1 is that it provides on the same partitioned feature space a non-deterministic but more efficient criterion for this difficult task.

3.2.1 Computation of the root $\pi^\star$

From Lemma 1, given a fixed temperature parameter $\lambda \in \Lambda(\bar{\pi})$, it is sufficient to compute $\pi^\star \in \mathcal{B}_\lambda(\bar{\pi})$ achieving $G(\pi^\star) = 0$ so that $\delta_{\pi^\star}^\lambda$ is an equalizer classifier. To compute this root $\pi^\star$, we propose to solve the following optimization problem

$$\pi^\star = \underset{\pi\in\mathbb{S}}{\operatorname{argmin}} \, \|G(\pi)\|_2^2. \tag{18}$$

As illustrated in Appendix C, this minimization problem is not necessary convex. Non-convex optimization has been actively studied in the last decades and several approaches such that gradient based algorithms as in (Ghadimi and Lan, 2013; Jain and Kar, 2017; Zhou et al., 2018; Fang et al., 2018; Li et al., 2021) or Monte-Carlo based algorithms like simulated annealing methods (Van Laarhoven and Aarts, 1987; Bertsimas and Tsitsiklis, 1993; Locatelli, 2000; Lecchini-Visintini et al., 2007) can be relevant for solving (18).

In practice, Monte-Carlo based algorithms using the Dirichlet distribution are able to converge efficiently to the global solution when the number of classes K is small enough. However, the complexity of this kind of method becomes too high when the number of classes becomes too large. In such a complex context, we propose in Appendix D an appropriate projected descent based algorithm for computing the priors $\pi^\star$ solution of (18).

3.2.2 Resulting classifier

Finally, from equations (8) and (11), our softmin discrete minimax classifier $\delta_{\pi^\star}^\lambda$ associated with the priors $\pi^\star$ solution of (18) assigns the class label $l \in \mathcal{Y}$ with probability

$$\hat{\mathbb{P}}\left(\delta_{\pi^\star}^\lambda(X_i) = l\right) \;=\; \frac{\exp\left[-\lambda \sum_{t\in\mathcal{T}} \sum_{k\in\mathcal{Y}} L_{kl}\, \pi_k^\star\, \hat{p}_{kt}\, \mathbb{1}_{\{\phi(X_i)=t\}}\right]}{\sum_{q=1}^K \exp\left[-\lambda \sum_{t\in\mathcal{T}} \sum_{k\in\mathcal{Y}} L_{kq}\, \pi_k^\star\, \hat{p}_{kt}\, \mathbb{1}_{\{\phi(X_i)=t\}}\right]}. \tag{19}$$

4 Experiments

As we emphasized in Introduction, our new criterion can be easily considered as a self-sufficient classifier or can be coupled with any pretrained decision trees or any pretrained Convolutional Neural Networks (CNNs) for addressing the previously studied prior issues. The experiments in Figure 1 and in Figure 2 illustrated the adjustment of a pretrained decision tree using our criterion. We first retrieve the leaves of the initial tree and we perform our softmin discrete minimax decision rule on this tree partitioning. In Subsection 4.1 we perform our new criterion as a self-sufficient classifier on two real databases and we compare our method to famous state of the art approaches. Then in Subsection 4.2 we explain how our new criterion can easily adjust pretrained CNNs for image classification tasks. Our code in Python is available at cypgilet.

4.1 Self-sufficient classifier for mixed features

Our new criterion can be considered as a self-sufficient classifier, in other words it can be easily applied directly to numeric or discrete or mixed attributes. For illustrating this fact, we consider two well known databases (Diabete (Johannes, 1988) and Scania Trucks (Scania, 2016)) coming from different real application domains. These two databases present different levels of difficulties depending on the class proportions, the loss function, the number of features and the number of instances. An overview of the main characteristics of each database is provided in Table 1 and a detailed description of these databases is available in Appendix E.2.

Table 1: Overview on each database. The number of instances is denoted by m and the number of attributes by d. Among the d features, d_n corresponds to the number of numeric attributes. Moreover, Stl denotes the loss function provided by the experts of the application domain in (Scania, 2016), such that $L_{12} = 10$, $L_{21} = 500$, and $L_{11} = L_{22} = 0$. Finally, $\hat{\pi}$ denotes the class proportions in each database.

DATABASE	m	d	d_n	K	$\hat{\pi}$	L
DIABETES (JOHANNES, 1988)	768	8	8	2	$[0.65, 0.35]$	$L_{0\text{-}1}$
SCANIA (SCANIA, 2016)	69,309	130	130	2	$[0.99, 0.01]$	Stl

For each database we perform a cross-validation procedure and we compare our new classifier with three common approaches adapted to deal with imbalanced datasets: the Weighted Logistic Regression (WLR), the Weighted Decision Tree (WDT), and the Discrete Minimax classifier (Gilet et al., 2020) (DMC). The WLR and the WDT are fitted using the algorithms provided by Scikit-Learn (Pedregosa et al., 2011) and their class-weights parameters are defined inversely proportional to class frequencies. At each iteration of the cross-validation procedure, we performed the DMC and our Softmin Discrete Minimax Classifier (SoftminDMC) on the same partitioned feature space (using a Tree partitioning for the Diabete database and the Kmeans algorithm for the Scania Trucks dataset). Regarding the calibration of the optimal number T of discrete profiles, we used the procedure provided in (Gilet et al., 2020).

The results are presented in Table 2. We observe that our SoftminDMC performs better than the other approaches (and than the DMC as we scoped in this paper) for minimizing the maximum of the class-conditional risks and for equalizing these risks per class. These results are highlighted on the difficult Scania-Trucks databases for which the SoftminDMC and the DMC allow to divide the risk of missing a failure (the class of interest) by more than three compared to the WDT and the WLR.

Table 2: Average results on each database. The notation δ (R) means that the classifier δ was applied on the real features. The notation δ (D) means that the classifier δ was performed on the partitionned version of each database. The results are presented as $[\text{mean} \pm \text{std}]$. For each criterion, the green font represents the best performances while the red font corresponds to the worst results. The criterion $\psi(\delta) = \max_{k\in\mathcal{Y}} \hat{R}_k(\delta) - \min_{k\in\mathcal{Y}} \hat{R}_k(\delta)$ measures how well the classifier δ is equalizer.

CRITERIA	CLASSIFIERS $\delta \in \Delta$	DIABETE Train	DIABETE Test	SCANIA-TRUCK Train	SCANIA-TRUCK Test
$\hat{r}(\delta)$	WLR (R)	0.31 ± 0.02	0.32 ± 0.03	0.70 ± 0.02	0.84 ± 0.13
	WDT (R)	0.27 ± 0.02	0.30 ± 0.04	0.69 ± 0.01	0.86 ± 0.05
	DMC (D)	0.23 ± 0.02	0.26 ± 0.01	8.95 ± 0.79	9.00 ± 0.76
	SoftminDMC (D)	0.26 ± 0.02	0.29 ± 0.01	4.81 ± 0.42	4.83 ± 0.43
$\max_{k\in\mathcal{Y}} \hat{R}_k(\delta)$	WLR (R)	0.33 ± 0.02	0.35 ± 0.02	37.7 ± 2.68	51.2 ± 16.9
	WDT (R)	0.37 ± 0.04	0.37 ± 0.04	17.7 ± 5.70	31.9 ± 8.79
	DMC (D)	0.45 ± 0.09	0.50 ± 0.04	9.05 ± 0.80	9.16 ± 0.74
	SoftminDMC (D)	0.26 ± 0.02	0.32 ± 0.01	5.06 ± 0.49	6.91 ± 1.61
$\psi(\delta)$	WLR (R)	0.08 ± 0.03	0.10 ± 0.05	37.4 ± 2.68	50.9 ± 16.9
	WDT (R)	0.27 ± 0.06	0.18 ± 0.05	17.2 ± 5.77	31.5 ± 8.86
	DMC (D)	0.34 ± 0.10	0.37 ± 0.06	9.05 ± 0.80	5.03 ± 3.66
	SoftminDMC (D)	0.01 ± 0.01	0.05 ± 0.01	0.41 ± 0.35	2.83 ± 1.35

4.2 Adjusting pretrained CNNs for prior issues

This subsection is now devoted to explain how our new criterion can adjust pretrained CNNs for image classification tasks. Let $\Phi : \mathcal{X} \to \mathcal{Y}$ denote a CNN (Goodfellow et al., 2016) which assigns a class label to each image $X \in \mathcal{X}$. Basically, the architecture of a CNN Φ composed of s hidden layers $h_1, \ldots, h_s$ can be modeled as

$$\Phi(X) \;=\; \delta \circ h_s \circ \cdots \circ h_1(X) \;=\; \delta \circ \varphi(X), \tag{20}$$

where $\delta(\cdot)$ denotes the output layer classifier and $\varphi(X) = h_s \circ \cdots \circ h_1(X)$ is the output of the last hidden layer (commonly called deep features). Usually in a CNN, the output decision rule δ aims to approximate the Bayes classifier and the softmax decision rule is often used to carry out this approximation (Goodfellow et al., 2016).

In this paper, we consider that the CNN Φ is already trained on a training set. From equation (1), its average empirical risk is given by

$$\hat{r}(\Phi) = \frac{1}{m}\sum_{i\in\mathcal{I}} L(Y_i, \Phi(X_i)) = \frac{1}{m}\sum_{i\in\mathcal{I}} L(Y_i, \delta\circ\varphi(X_i)) \tag{21}$$

$$= \underbrace{\frac{1}{m}\sum_{i\in\mathcal{I}} L(Y_i, \delta(Z_i))}_{:=\hat{r}_\varphi(\delta)}. \tag{22}$$

where $Z_i = \varphi(X_i)$ are the deep features of the image X_i. In other words, the average risk $\hat{r}(\Phi)$ of a CNN Φ is equal to the empirical risk $\hat{r}_\varphi(\delta)$ of the decision rule δ applied on the deep features.

From equations (2) and (5), this average risk can be decomposed regarding the priors as follows

$$\hat{r}(\pi, \Phi) = \sum_{k\in\mathcal{Y}} \pi_k \hat{R}_k(\Phi) = \hat{r}_\varphi(\pi, \delta) = \sum_{k\in\mathcal{Y}} \pi_k \hat{R}_k(\delta). \tag{23}$$

In other words, the sensitivity of a trained CNN Φ to imbalanced classes and prior probability shifts comes from the sensitivity of the output decision rule δ to these prior issues.

We therefore propose here to replace the output decision rule δ of the trained CNN (which is common in the literature as in (Gilet et al., 2020; Tian et al., 2020)) with our Softmin Discrete Minimax Classifier (SoftminDMC). The steps to easily couple any pretrained CNN with our output SoftminDMC are summarized as follows and the architecture of the resulting adjusted CNN is illustrated in Figure 3.

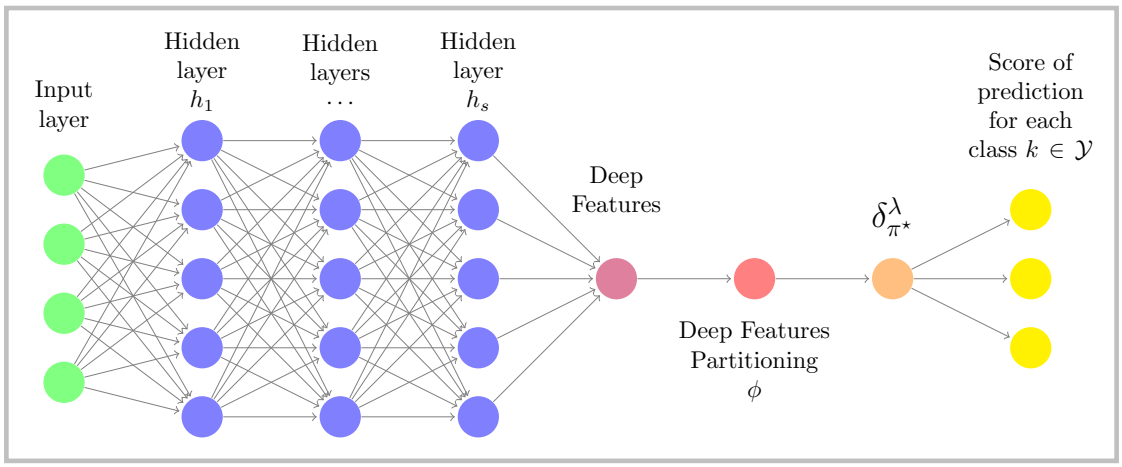

Figure 3: Scheme of our coupling method to adjust pretrained CNNs using our output Softmin Discrete Minimax Classifier. The probability score of prediction for each class $k \in \mathcal{Y}$ is calculating using (19).

Step 1. We first retrieve the hidden layers of the pretrained CNN which were fitted on training samples. These hidden layers allow to compute the deep features associated with each input image.

Step 2. We then add a new layer ϕ which aims to partition the deep feature space. In practice, an easy way is to consider the Kmeans partitioning and to set the number T of centroids (each centroid corresponding to a discrete profile) such that the discrete Bayes classifier $\delta^{\mathrm{B}}_{\hat{\pi}}$ [given by (7) when considering the priors $\hat{\pi}$] and the softmax output decision rule achieve both similar average risks (1) (if possible on new learning samples in order to avoid overfitting).

Step 3. We then compute the priors $\pi^\star$ solution of (18) using Algorithm 1 or Algorithm 2 provided in Appendix D.

Step 4. Our output Softmin Discrete Minimax decision rule $\delta^{\lambda}_{\pi^\star}$ is finally given by (19). It allows to adjust the initial CNN face to imbalanced class-conditional risks and prior probability shifts. It moreover provides the probability score of prediction for each class $k \in \mathcal{Y}$.

4.2.1 Experiments on real medical databases

We consider here three real medical databases (*OCTMNIST, DermaMNIST, BreastMNIST*) (Yang et al., 2021, 2020) which differ in the number of images, the number of classes and the class proportions. Each database contains a training set, a validation set and a test set with 28×28 images. Table 3 summarizes the main characteristics of each database and a detailed overview of each databases is provided in Appendix E.2.

Table 3: Overview of the real medical databases: 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}	$\pi^{train} = \pi^{val}$	π^{test}
DermaMNIST	7	7,007	1,003	2,005	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
OCTMNIST	4	97,477	10,832	1,000	Min = 0.08 Max = 0.47	Min = 0.25 Max = 0.25

In order to illustrate that our approach can be coupled with any kind of CNN architecture, we considered the CNN *ResNet-18* (He et al., 2016) for the DermaMNIST and BreastMNIST databases and the CNN *EfficientNet-B7* (Tan and Le, 2019) for the OCTMNIST database. We trained each CNN on the training set with 100 epochs using the cross-entropy loss and a SGD optimizer as in (Yang et al., 2020).

For each pretrained CNN we compared four output layer classifiers: the initial softmax decision rule (CNN-Softmax), the interesting imbalance calibration approach (Tian et al., 2020) (CNN-IC) which is designed to be applied on pretrained CNNs too, the Discrete Minimax Classifier (Gilet et al., 2020) (CNN-DMC) and our Softmin discrete minimax classifier (CNN-SoftminDMC). Each output classifier was fitted on the deep features associated with the validation set in order to avoid overfitting possibly due to the deep features coming from the training set. Finally, the generalization performances were evaluated on each test set.

For each output layer classifier δ, we compare the average risk (1), the maximum of the class-conditional risks (3) and the gap between the maximum and the minimum of these risks per class. The results are presented in Table 4. Since our approach is not fitted by minimizing the average risk (1), it is not expected to outperform the other methods regarding this criterion, but it remains closed to the CNN-DMC ones, which illustrates Theorem 1. As it was the objective of this paper, our method outperforms all the other output decision rules for minimizing the maximum of the class-conditional risks and for equalizing these risks per class.

Table 4: Results of each output classifier on the real medical databases. For each criterion, the green font represents the best performances while the red font corresponds to the worst results. The criterion $\psi(\delta) = \max_{k \in \mathcal{Y}} \hat{R}_k(\delta) - \min_{k \in \mathcal{Y}} \hat{R}_k(\delta)$ measures how well the classifier δ is equalizer.

CRITERIA	**CLASSIFIERS** $\delta \in \Delta$	**DermaMNIST** Val	**DermaMNIST** Test	**OctMNIST** Val	**OctMNIST** Test	**BreastMNIST** Val	**BreastMNIST** Test
$\hat{r}(\delta)$	CNN-Softmax	0.29	0.30	0.07	0.28	0.16	0.16
	CNN-IC	0.37	0.39	0.08	0.26	0.15	0.15
	CNN-DMC	0.56	0.58	0.09	0.27	0.15	0.15
	CNN-SoftminDMC	0.57	0.59	0.21	0.31	0.19	0.17
$\max_{k \in \mathcal{Y}} \hat{R}_k(\delta)$	CNN-Softmax	1.00	1.00	0.41	0.72	0.43	0.50
	CNN-IC	0.99	1.00	0.37	0.65	0.24	0.29
	CNN-DMC	0.65	0.91	0.28	0.54	0.19	0.24
	CNN-SoftminDMC	0.64	0.87	0.24	0.50	0.19	0.19
$\psi(\delta)$	CNN-Softmax	0.83	0.84	0.38	0.69	0.36	0.46
	CNN-IC	0.70	0.66	0.32	0.62	0.12	0.18
	CNN-DMC	0.23	0.43	0.22	0.50	0.05	0.11
	CNN-SoftminDMC	0.15	0.32	0.11	0.32	0.00	0.02

4.2.2 Experiments on the CIFAR-100 database

In order to illustrate that our criterion can easily deal with a large number of classes, we consider a new experiment on the famous CIFAR-100 database (Krizhevsky, 2009) which contains $K = 100$ classes. While this well known

visual recognition database presents perfectly equal class proportions, the major difficulty appearing is that the class-conditional risks result imbalanced for most classifiers. The scope of this experiment is to illustrate Theorem 1 in such a large scale database.

For this experiment we considered the deep features extracted from the last hidden layer of the CNN EfficientNet-B0 (Tan and Le, 2019) and we considered the $L_{0\text{-}1}$ loss function. The other settings for this experiment are provided in Appendix E.1. We partitioned the deep features using the Kmeans procedure with $T = 1200$ centroids (each centroid corresponding to a discrete profile) and we compared on the same partitioned deep feature space the Discrete Bayes Classifier (CNN-DBC) $\delta^{\mathrm{B}}_{\hat{\pi}}$ [given by (7) when considering the priors $\hat{\pi}$], the Discrete Minimax Classifier $\delta^{\mathrm{B}}_{\bar{\pi}}$ [given by (7) when considering the priors $\bar{\pi} = \operatorname{argmax}_{\pi \in \mathbb{S}} V_B(\pi)$] and our Softmin Discrete Minimax Classifier $\delta^{\lambda}_{\pi^\star}$. We chose to compare these three algorithms in order to illustrate the impact of both the priors $\hat{\pi}, \bar{\pi}, \pi^\star \in \mathbb{S}$ and the randomization on the same deep feature space partitioning. Regarding our Softmin Discrete Minimax Classifier $\delta^{\lambda}_{\pi^\star}$, we computed the priors $\pi^\star$ solution of (18) using Algorithm 2 in Appendix D.

The results are displayed in Figure 4. Let us note that the Discrete Bayes Classifier leaded to similar results than the initial CNN-Softmax. We can observe that its class-conditional risks are highly imbalanced. The DMC ouptput classifier balances better these risks per class. Our new Sotmin-DMC criterion got the best results in this difficult task and achieved $\max_{k \in \mathcal{Y}} \hat{R}_k(\delta^{\lambda}_{\pi^\star}) \leq \max_{k \in \mathcal{Y}} \hat{R}_k(\delta^{B}_{\bar{\pi}})$, which especially illustrates Theorem 1. Additional experiments are provided in Appendix E.1 when considering $T = 800$ discrete profiles and highlight these results.

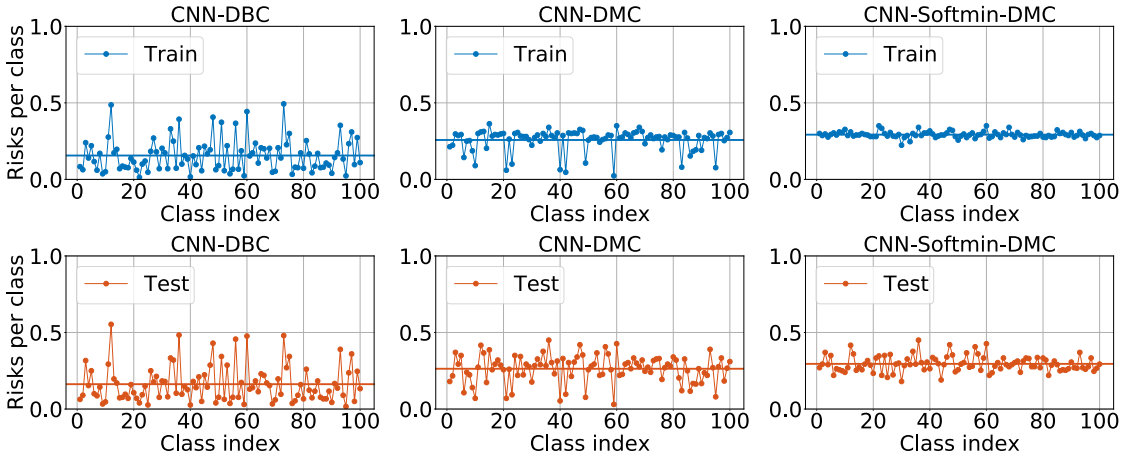

Figure 4: CIFAR-100 database: Class-conditional risks associated with the CNN-DBC, CNN-DMC and CNN-Softmin-DMC classifiers on both the training and test datasets.

5 Conclusion

This paper proposes a softmin discrete minimax classifier which belongs to the field of randomized minimax decision rules for supervised classification tasks. Our new approach aims to address the issues of imbalanced class-conditional risks and prior probability shifts. Our new classifier converges to the deterministic discrete minimax decision rule and seeks to perform better for minimizing the maximum of the class-conditional risks. It can be considered as a self-sufficient classifier or can be coupled with any pretrained CNNs or decision trees.

An important asset of our new minimax classifier is to provide probability scores of prediction for each class, which are for instance important in visual recognition applications. In our opinion, this asset can open an interesting path to develop a minimax classifier for multi-labels decision-making (Zhang and Zhou, 2013; Xu et al., 2019) when dealing with imbalanced datasets and prior probability shifts. Moreover, this new criterion could be easily adapted to a Box-constrained minimax classifier and a minimax regret decision rule, which were both studied in the context of partitioned feature spaces in (Gilet et al., 2020; Gilet, 2021). Ongoing research are devoted to study the generalization error of our new minimax classifier based on the feature space partitioning. It would be moreover interesting to connect and extend our research to the recently studied minimax fairness area (Martinez et al., 2020; Diana et al., 2021) which aims to achieve balanced/fair predictions by sensitive attribute groups.

References

Ávila Pires, B., Szepesvari, C., and Ghavamzadeh, M. (2013). Cost-sensitive multiclass classification risk bounds. In *Proceedings of the 30th International Conference on Machine Learning*.

Berger, J. O. (1985). *Statistical decision theory and Bayesian analysis; 2nd ed.* Springer Series in Statistics. Springer, New York.

Bertsimas, D. and Tsitsiklis, J. (1993). Simulated annealing. *Statistical Science*, 8(1):10–15.

Borovkov, A. A. (1998). *Mathematical Statistics*. Gordon and Breach Sciences Publishers, Amsterdam.

Braga-Neto, U. and Dougherty, E. R. (2005). Exact performance of error estimators for discrete classifiers. *Elsevier Pattern Recognition*, 38(11):1799–1814.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees*. Chapman and Hall/CRC, 1st edition.

Buda, M., Maki, A., and Mazurowski, M. A. (2018). A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*.

Cao, K., Wei, C., Gaidon, A., Arechiga, N., and Ma, T. (2019). Learning imbalanced datasets with label-distribution-aware margin loss. In *Advances in Neural Information Processing Systems*, volume 32, pages 1567–1578.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

Chawla, N. V., Lazarevic, A., Hall, L. O., and Bowyer, K. W. (2003). Smoteboost: Improving prediction of the minority class in boosting. In *Knowledge Discovery in Databases: PKDD 2003*.

Colliot, O. and Burgos, N. (2020). Machine learning for classification and prediction of brain diseases: recent advances and upcoming challenges. *Current Opinion in Neurology*, 33:439–450.

Cui, Y., Jia, M., Lin, T.-Y., Song, Y., and Belongie, S. (2019). Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277.

Dalton, L. A. and Dougherty, E. R. (2011). 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*, 59:115–129.

Devroye, L., Györfi, L., and Lugosi, G. (1996). *A Probabilistic Theory of Pattern Recognition*. Springer-Verlag New York, 2nd edition.

Diana, E., Gill, W., Kearns, M., Kenthapadi, K., and Roth, A. (2021). Minimax group fairness: Algorithms and experiments. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 66–76.

Dong, Q., Gong, S., and Zhu, X. (2019). Imbalanced deep learning by minority class incremental rectification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Drummond, C. and C. Holte, R. (2003). C4.5, class imbalance, and cost sensitivity: Why under-sampling beats oversampling. *Proceedings of the ICML'03 Workshop on Learning from Imbalanced Datasets*.

Duda, R. O., Hart, P. E., and Stork, D. G. (2000). *Pattern Classification*. John Wiley and Sons, 2nd edition.

Elkan, C. (2001). The foundations of cost-sensitive learning. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence - Volume 2*, pages 973–978.

Fang, C., Li, C. J., Lin, Z., and Zhang, T. (2018). Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. *Advances in Neural Information Processing Systems*, 31.

Ferguson, T. (1967). *Mathematical Statistics : A Decision Theoretic Approach*. Academic Press.

Fonda, A. and Gidoni, P. (2016). Generalizing the Poincaré–Miranda Theorem: the avoiding cones condition. *Annali di Matematica Pura ed Applicata (1923-)*, 195(4):1347–1371.

Frankowska, H. (2018). The Poincaré–Miranda Theorem and viability condition. *Journal of Mathematical Analysis and Applications*, 463(2):832–837.

Gentile, C., Li, S., and Zappella, G. (2014). Online clustering of bandits. In Xing, E. P. and Jebara, T., editors, *Proceedings of Machine Learning Research*, volume 32, pages 757–765, Bejing, China. PMLR.

Ghadimi, S. and Lan, G. (2013). Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368.

Gilet, C. (2021). *Discrete minimax classifier for personalized diagnosis in medicine*. PhD Thesis, Université Côte d'Azur.

Gilet, C., Barbosa, S., and Fillatre, L. (2019). Minimax classifier with box constraint on the priors. In *Machine Learning for Health (ML4H) at NeurIPS 2019*. Proceedings of Machine Learning Research.

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

Gilet, C., Guyomard, M., Barbosa, S., and Fillatre, L. (2020). Adjusting Decision Trees for Uncertain Class Proportions. In *Workshop on Uncertainty in Machine Learning at ECML/PKDD 2020*.

González, P., Castaño, A., Nitesh, C., and Del Coz, J. J. (2017). A review on quantification learning. *ACM Computing Surveys*.

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

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

Hastie, T., Tibshirani, R., and Friedman, J. (2009). *The Elements of Statistical Learning*. Springer-Verlag New York, 2nd edition.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, pages 1263–1284.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. *CVPR*, pages 770–778.

Jain, P. and Kar, P. (2017). Non-convex optimization for machine learning. *Foundations and Trends in Machine Learning*, 10(3-4):142–363.

Japkowicz, N. and Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis*, pages 429–449.

Johannes, R. S. (1988). Using the adap learning algorithm to forecast the onset of diabetes mellitus. *Johns Hopkins APL Technical Digest*, 10:262–266.

Krizhevsky, A. (2009). Learning multiple layers of features from tiny images. https://www.cs.toronto.edu/ kriz/cifar.html.

Kukar, M. and Kononenko, I. (1998). Cost-sensitive learning with neural networks. *European Conference on Artificial Intelligence*.

Kulpa, W. (1997). The Poincaré-Miranda Theorem. *The American Mathematical Monthly*, 104(6):545–550.

Lawrence, S., Burns, I., Back, A., Tsoi, A. C., and Giles, C. L. (1998). *Neural Network Classification and Prior Class Probabilities*. Springer Berlin Heidelberg.

Lecchini-Visintini, A., Lygeros, J., and Maciejowski, J. (2007). Simulated annealing: Rigorous finite-time guarantees for optimization on continuous domains. *Advances in Neural Information Processing Systems (NeurIPS 2007)*, 20.

Lee, H.-j. and Cho, S. (2006). The novelty detection approach for different degrees of class imbalance. In King, I., Wang, J., Chan, L.-W., and Wang, D., editors, *Neural Information Processing*. Springer Berlin Heidelberg.

Li, S., Gentile, C., and Karatzoglou, A. (2016). Graph clustering bandits for recommendation. *arXiv:1605.00596*.

Li, Z., Bao, H., Zhang, X., and Richtárik, P. (2021). Page: A simple and optimal probabilistic gradient estimator for nonconvex optimization. In *International Conference on Machine Learning*, pages 6286–6295. PMLR.

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B., and Sánchez, C. I. (2017). A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88.

Locatelli, M. (2000). Simulated annealing algorithms for continuous global optimization: convergence conditions. *Journal of Optimization Theory and applications*, 104(1):121–133.

MacQueen, J. (1967). Some methods for classification and analysis of multivariate observations. *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297.

Martinez, N., Bertran, M., and Sapiro, G. (2020). Minimax pareto fairness: A multi objective perspective. In *International Conference on Machine Learning*. PMLR.

Mawhin, J. (2007). Le Théoreme du point fixe de Brouwer: un siecle de métamorphoses. *Sci. Tech. Perspect*, 2(10):1–2.

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

Moreno-Torres, J. G., Raeder, T., Alaiz-Rodríguez, R., Chawla, N. V., and Herrera, F. (2012). A unifying view on dataset shift in classification. *Pattern Recognition*.

Pedregosa, F., Varoquaux, G., Gramfort, A., and et al (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Poor, H. V. (1994). *An Introduction to Signal Detection and Estimation*. Springer-Verlag New York, 2nd edition.

Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., and Lawrence, N. D. (2008). *Dataset Shift in Machine Learning*. MIT Press.

Rao, C. R. (1973). *Linear Statistical Inference and its Applications*. Wiley.

Scania, C. A. (2016). APS failure at Scania Trucks data set. https://www.kaggle.com/uciml/aps-failure-at-scania-trucks-data-set.

Scott, C. and Nowak, R. D. (2006). Minimax-optimal classification with dyadic decision trees. *IEEE Transactions on Information Theory*, 52(4).

Tan, M. and Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International Conference on Machine Learning*.

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

Van Laarhoven, P. J. and Aarts, E. H. (1987). Simulated annealing. In *Simulated annealing: Theory and applications*, pages 7–15. Springer.

Vapnik, V. (1999). An overview of statistical learning theory. *IEEE transactions on Neural Networks*, 10 5:988–99.

Xu, D., Shi, Y., Tsang, I. W., Ong, Y.-S., Gong, C., and Shen, X. (2019). Survey on multi-output learning. *IEEE transactions on neural networks and learning systems*, 31(7):2409–2429.

Xu, Z., Dan, C., Khim, J., and Ravikumar, P. (2020). Class-weighted classification: Trade-offs and robust approaches. In *International Conference on Machine Learning*, pages 10544–10554. PMLR.

Yablon, M. and Chu, J. T. (1982). Approximations of bayes and minimax risks and the least favorable distribution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 35–40.

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

Yang, J., Shi, R., and Ni, B. (2020). *MedMNIST* databases. https://zenodo.org/record/4269852#.X_mdsulKiHE.

Yang, J., Shi, R., Wei, D., Liu, Z., Zhao, L., Ke, B., Pfister, H., and Ni, B. (2021). Medmnist v2: A large-scale lightweight benchmark for 2d and 3d biomedical image classification. *arXiv:2110.14795*.

Zhang, M.-L. and Zhou, Z.-H. (2013). A review on multi-label learning algorithms. *IEEE transactions on knowledge and data engineering*, 26(8):1819–1837.

Zhou, D., Xu, P., and Gu, Q. (2018). Stochastic nested variance reduction for nonconvex optimization. *Advances in Neural Information Processing Systems*, 31.

Supplementary Material

Softmin Discrete Minimax Classifier for Imbalanced Datasets and Prior Probability Shifts

A Example of prior probability shifts

We consider an experiment for which we generated a training dataset containing $m = 5,000$ instances described by $d = 2$ features, coming from $K = 2$ classes and such that the class proportions satisfy $\hat{\pi} = [0.90, 0.10]$. To this aim, we used the `make_blobs` function provided by Scikit-Learn. Figure 5, Left, displays the scatter plot of the learning instances. Let us note that this is the same dataset as the one presented in Figure 1 in the main paper.

For this experiment, we first fitted a Logistic Regression decision rule $\delta^{LR} \in \mathcal{D}$ on this training set. Figure 5, Left, displays the decision boundary associated with the fitted Logistic Regression. As reported in Figure 5, Right, the Logistic Regression δ^{LR} achieved an average risk $\hat{r}\left(\delta^{LR}\right) = 0.075$ which seems highly satisfying. But if we focus on the class-conditional risks, we observe that 50% of the instances from class C2 were misclassified. As reminded in the introduction, this issue comes from the imbalanced class-proportions on the training set and from the fact that the classes are not easily separable on the feature space.

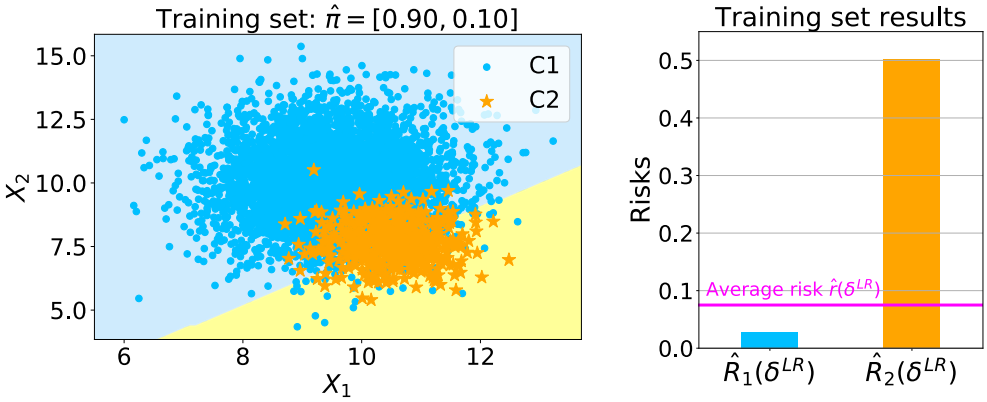

Figure 5: Results associated with the Logistic Regression δ^{LR} on the training set.

In order to illustrate the issues of prior probability shifts, we then applied the fitted Logistic Regression δ^{LR} on 5 different test datasets containing $m' = 1,000$ instances. Each test dataset was generated from the same feature distributions in each class, but these test sets differ according to the class proportions $\pi' = [\pi'_1, \pi'_2]$ ranging over the simplex $\mathbb{S}$. Figure 6 displays the scatter plots of each test dataset and their associated class-proportions π'. The last subfigure of Figure 6 describes the average risks associated with each test dataset. Since we have $K = 2$ classes, theses average risks (5) can be written as

$$\hat{r}\left(\pi', \delta^{LR}\right) = \pi'_1[\hat{R}_1\left(\delta^{LR}\right) - \hat{R}_2\left(\delta^{LR}\right)] + \hat{R}_2\left(\delta^{LR}\right). \tag{24}$$

As reminded in Introduction, this is a linear function with respect to π'_1 between the class-conditional risks $\left\{\hat{R}_1\left(\delta^{LR}\right), \hat{R}_2\left(\delta^{LR}\right)\right\}$. Since these risks per class were highly imbalanced on the training set (see Figure 5), it follows that the average risk (24) of this fitted Logistic Regression δ^{LR} is highly sensitive to prior probability shifts.

To conclude this part, minimizing the maximum of the class-conditional risks (which generally leads to balance these risks per class) would make the classifier robust to prior probability shifts since the average risk (24) would remain almost constant. This is the scope and the asset of the minimax criterion as presented in (Ferguson, 1967; Berger, 1985; Poor, 1994; Gilet et al., 2020). This is moreover the objective of our present paper, as illustrated in Figure 2 in the main paper.

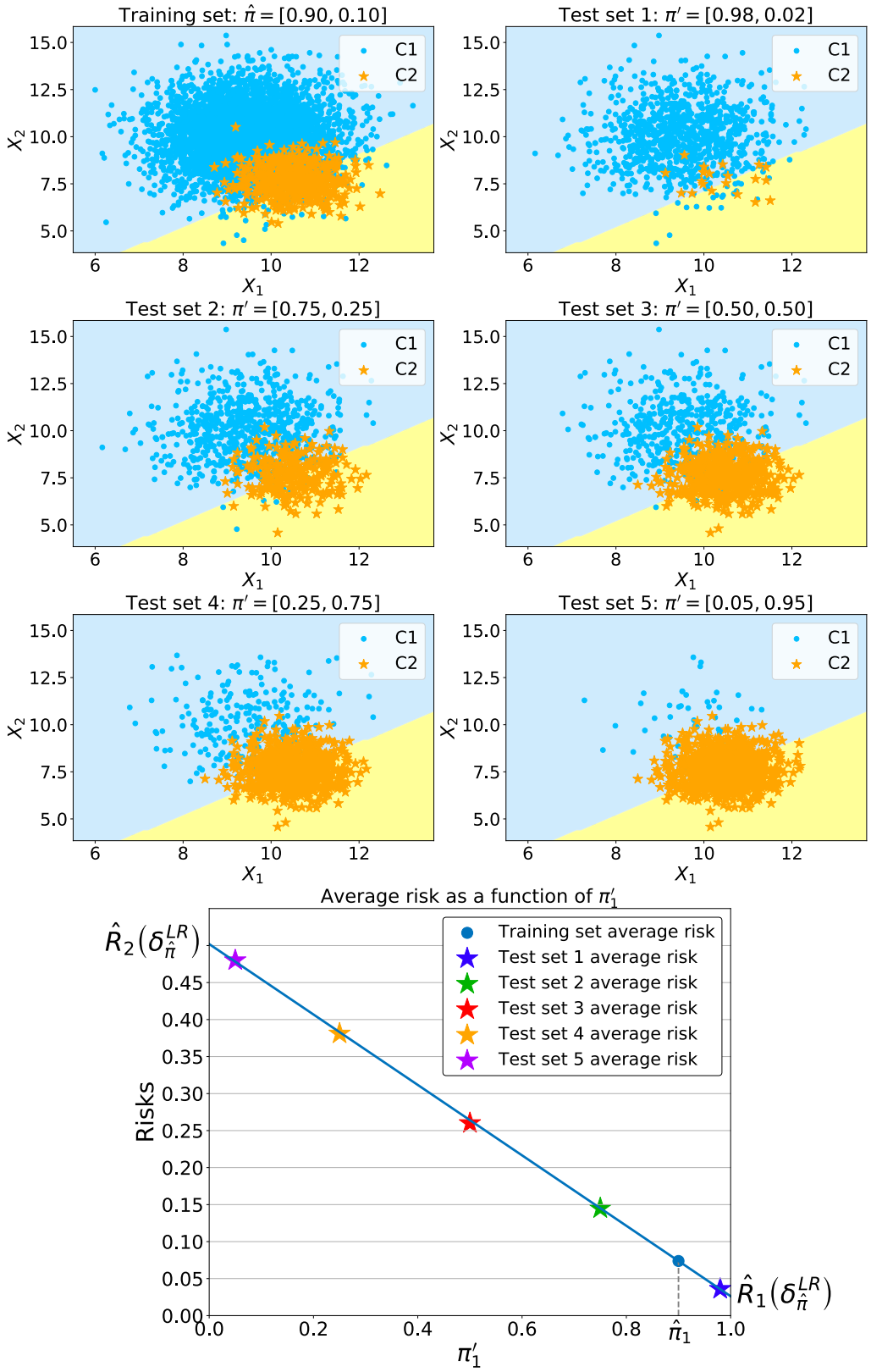

Figure 6: Illustration of prior probability shifts for $K = 2$ classes.

B Proves of the paper

B.1 Detailed proof of Proposition 1

The following lemma estimates analytically the risk of our softmin discrete classifier $\delta_{\hat{\pi}}^{\lambda}$ on the learning set $\mathcal{S}$ with respect to the training class proportions $\hat{\pi} \in \mathbb{S}$.

Lemma 2. *For any fixed parameter $\lambda > 0$, the average risk of our softmin discrete classifier $\delta_{\hat{\pi}}^{\lambda}$ associated with the class proportions of the training set $\hat{\pi} \in \mathbb{S}$ is given by*

$$\hat{r}\left(\delta_{\hat{\pi}}^{\lambda}\right) \;=\; \sum_{k\in\mathcal{Y}} \sum_{t\in\mathcal{T}} \sum_{l\in\mathcal{Y}} \hat{\pi}_k \, L_{kl} \, \hat{p}_{kt} \, \frac{e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jl}\, \hat{\pi}_j \, \hat{p}_{jt}}}{\sum_{q=1}^{K} e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jq}\, \hat{\pi}_j \, \hat{p}_{jt}}}. \tag{25}$$

Proof. Since $\delta_{\hat{\pi}}^{\lambda} \in \Delta$, we have from equations (2) and (3)

$$\begin{aligned} \hat{r}\left(\delta_{\hat{\pi}}^{\lambda}\right) \;&=\; \sum_{k\in\mathcal{Y}} \hat{\pi}_k \sum_{l\in\mathcal{Y}} L_{kl}\, \hat{\mathbb{P}}\left(\delta_{\hat{\pi}}^{\lambda}(X_i) = l \mid Y_i = k\right) \\ &=\; \sum_{k\in\mathcal{Y}} \sum_{l\in\mathcal{Y}} \hat{\pi}_k \, L_{kl}\, \hat{\mathbb{P}}\left(\delta_{\hat{\pi}}^{\lambda}(X_i) = l \mid Y_i = k\right). \end{aligned} \tag{26}$$

Let us define $\xi : \mathcal{T} \to \mathcal{Y}$ such that for all $l \in \mathcal{Y}$

$$\hat{\mathbb{P}}\left(\xi(t) = l\right) \;=\; \frac{e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jl}\, \hat{\pi}_j \, \hat{p}_{jt}}}{\sum_{q=1}^{K} e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jq}\, \hat{\pi}_j \, \hat{p}_{jt}}}. \tag{27}$$

Then, from equation (11), from the quantities $\mathcal{F} = \left\{f_1(X_i), \ldots, f_K(X_i)\right\}$ defined in equation (7), and from the Law of total probability, it follows that for all $l \in \mathcal{Y}$ and all $k \in \mathcal{Y}$

$$\begin{aligned} \hat{\mathbb{P}}\left(\delta_{\hat{\pi}}^{\lambda}(X_i) = l \mid Y_i = k\right) \;&=\; \sum_{t\in\mathcal{T}} \hat{\mathbb{P}}\left(\phi(X_i) = t,\, \xi(t) = l \mid Y_i = k\right) \\ &=\; \sum_{t\in\mathcal{T}} \hat{\mathbb{P}}\left(\phi(X_i) = t \mid Y_i = k\right) \hat{\mathbb{P}}\left(\xi(t) = l \mid Y_i = k\right) \\ &=\; \sum_{t\in\mathcal{T}} \hat{p}_{kt}\, \hat{\mathbb{P}}\left(\xi(t) = l \mid Y_i = k\right) \\ &=\; \sum_{t\in\mathcal{T}} \hat{p}_{kt}\, \hat{\mathbb{P}}\left(\xi(t) = l\right). \end{aligned} \tag{28}$$

Hence, from equations (26), (27) and (28), we finally obtain the result (25). □

Proof of Proposition 1. Under Assumption 1, the probabilities $\hat{p}_{kt}$ defined in (6) are considered fixed. Hence, the average risk (25) considered as a function of the priors $\pi \in \mathbb{S}$ is

$$\begin{aligned} &V_{\lambda} :\; \mathbb{S} \to [0, +\infty) \\ &\qquad \pi \mapsto \sum_{k\in\mathcal{Y}} \sum_{t\in\mathcal{T}} \sum_{l\in\mathcal{Y}} \pi_k \, L_{kl} \, \hat{p}_{kt} \, \frac{e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jl}\, \pi_j \, \hat{p}_{jt}}}{\sum_{q=1}^{K} e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jq}\, \pi_j \, \hat{p}_{jt}}}. \end{aligned} \tag{29}$$

Then, from equation (2), this average risk can be rewritten as

$$V_{\lambda}(\pi) \;=\; \sum_{k\in\mathcal{Y}} \pi_k \, \hat{R}_k\left(\delta_{\pi}^{\lambda}\right), \tag{30}$$

where for all $k \in \mathcal{Y}$, the class-conditional risks $\hat{R}_k\left(\delta_{\pi}^{\lambda}\right)$ are analytically given by

$$\hat{R}_k\left(\delta_{\pi}^{\lambda}\right) \;=\; \sum_{t\in\mathcal{T}} \sum_{l\in\mathcal{Y}} L_{kl} \, \hat{p}_{kt} \, \frac{e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jl}\, \pi_j \, \hat{p}_{jt}}}{\sum_{q=1}^{K} e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jq}\, \pi_j \, \hat{p}_{jt}}}, \tag{31}$$

which proves Proposition 1. □

B.2 Detailed proof of Proposition 2

For the following, let us remind that

$$\Pi := \left\{ \pi \in \mathbb{S} : \forall t \in \mathcal{T}, \exists! \, l \in \mathcal{Y}, \; l = \operatorname*{argmin}_{q \in \mathcal{Y}} \sum_{j \in \mathcal{Y}} L_{jq} \, \pi_j \, \hat{p}_{jt} \right\}.$$

We first prove that for all $\pi \in \Pi$ the randomized softmin classifier δ_π^λ converges in probability to the discrete Bayes classifier δ_π^B as λ goes to infinity. Then, we prove that for all $\pi \in \mathbb{S}$ its average risk $V_\lambda(\pi)$ converges pointwise to the Bayes risk $V_B(\pi)$ as λ goes to infinity. For these two proves we will use the following lemma.

Lemma 3. *Let $a := [a_1, \ldots, a_K] \in \mathbb{R}_+^K$ and let $j = \operatorname{argmin}_{l \in \mathcal{Y}} a_l$ such that j is unique. Moreover, given $\lambda > 0$, let us define for all $k \in \mathcal{Y}$*

$$\sigma_k^\lambda(a) := \frac{e^{-\lambda \, a_k}}{\sum_{l=1}^K e^{-\lambda \, a_l}}. \tag{32}$$

Thence, for all $k \in \mathcal{Y}$,

$$\sigma_k^\lambda(a) \xrightarrow[\lambda \to +\infty]{} \mathbb{1}_{\left\{k = \operatorname{argmin}_{l \in \mathcal{Y}} a_l\right\}}. \tag{33}$$

Proof. Let $a \in \mathbb{R}_+^K$ and let $j \in \mathcal{Y}$ such that $j = \operatorname{argmin}_{l \in \mathcal{Y}} a_l$, which is unique. For all $k \in \mathcal{Y}$, we have

$$\sigma_k^\lambda(a) \; = \; \frac{e^{-\lambda \, a_k}}{\sum_{l=1}^K e^{-\lambda \, a_l}} \; = \; \frac{e^{-\lambda \, a_k}}{\sum_{l=1}^K e^{-\lambda \, a_l}} \cdot \frac{e^{\lambda \, a_j}}{e^{\lambda \, a_j}} \; = \; \frac{e^{-\lambda \, (a_k - a_j)}}{1 + \sum_{l \neq j} e^{-\lambda \, (a_l - a_j)}} \tag{34}$$

Since $a_j = \min_{l \in \mathcal{Y}} a_l$, it follows that for all $l \neq j$, $(a_l - a_j) > 0$, and thus $e^{-\lambda \, (a_l - a_j)} \to 0$ as $\lambda \to +\infty$. By considering this in (34), we get

$$\sigma_j^\lambda(a) \xrightarrow[\lambda \to +\infty]{} 1 \tag{35}$$

and for all $l \in \mathcal{Y}, l \neq j$,

$$\sigma_l^\lambda(a) \xrightarrow[\lambda \to +\infty]{} 0. \tag{36}$$

Hence, we obtain, for all $k \in \mathcal{Y}$,

$$\sigma_k^\lambda(a) \xrightarrow[\lambda \to +\infty]{} \mathbb{1}_{\left\{k = \operatorname{argmin}_{l \in \mathcal{Y}} a_l\right\}}. \tag{37}$$

□

Proof of the convergence in probability of δ_π^λ to δ_π^B for all $\pi \in \Pi$. Let $\pi \in \Pi$ fixed and let us consider $X_i \in \mathcal{X}$ an aleatory real feature profile. Let us remind that for all $q \in \mathcal{Y}$,

$$f_q(X_i) = \sum_{t \in \mathcal{T}} \sum_{k \in \mathcal{Y}} L_{kq} \, \pi_k \, \hat{p}_{kt} \, \mathbb{1}_{\{\phi(X_i) = t\}}.$$

Since $\pi \in \Pi$, it follows that $l = \operatorname{argmin}_{q \in \mathcal{Y}} f_q(X_i)$ is unique. In order to prove the convergence in probability of $\delta_\pi^\lambda(X_i)$ to $\delta_\pi^B(X_i)$ as λ goes to infinity, it is sufficient to prove that

$$\lim_{\lambda \to +\infty} \mathbb{P}\left(\delta_\pi^\lambda(X_i) = \delta_\pi^B(X_i)\right) \; = \; 1.$$

From the Law of total probability, we have

$$\begin{aligned} \lim_{\lambda \to +\infty} \mathbb{P}\left(\delta_\pi^\lambda(X_i) = \delta_\pi^B(X_i)\right) \; &= \; \lim_{\lambda \to +\infty} \sum_{l \in \mathcal{Y}} \mathbb{P}\left(\delta_\pi^\lambda(X_i) = l, \, \delta_\pi^B(X_i) = l\right) \\ &= \; \lim_{\lambda \to +\infty} \sum_{l \in \mathcal{Y}} \mathbb{P}\left(\delta_\pi^\lambda(X_i) = l\right) \, \mathbb{P}\left(\delta_\pi^B(X_i) = l\right). \end{aligned} \tag{38}$$

Let us note that the deterministic discrete Bayes classifier defined in equation (7) can also be viewed as a randomized decision rule such that for all $l \in \mathcal{Y}$,

$$\mathbb{P}\left(\delta_\pi^B(X_i) = l\right) \; = \; \mathbb{1}_{\left\{l = \operatorname{argmin}_{q \in \mathcal{Y}}\right\}}. \tag{39}$$

Moreover, from Definition 1, let us remind that for all $l \in \mathcal{Y}$,

$$\hat{\mathbb{P}}\left(\delta_\pi^\lambda(X_i) = l\right) \;=\; \frac{e^{-\lambda\, f_l(X_i)}}{\sum_{q\in\mathcal{Y}} e^{-\lambda\, f_q(X_i)}}. \tag{40}$$

Hence, replacing (39) and (40) in (38), it follows that

$$\lim_{\lambda\to+\infty} \mathbb{P}\left(\delta_\pi^\lambda(X_i) = \delta_\pi^B(X_i)\right) \;=\; \lim_{\lambda\to+\infty} \sum_{l\in\mathcal{Y}} \frac{e^{-\lambda\, f_l(X_i)}}{\sum_{q\in\mathcal{Y}} e^{-\lambda\, f_q(X_i)}}\, \mathbb{1}_{\left\{l=\operatorname{argmin}_{q\in\mathcal{Y}} f_q(X_i)\right\}}.$$

Thence, from Lemma 3, we finally get

$$\lim_{\lambda\to+\infty} \mathbb{P}\left(\delta_\pi^\lambda(X_i) = \delta_\pi^B(X_i)\right) \;=\; \lim_{\lambda\to+\infty} \sum_{l\in\mathcal{Y}} \frac{e^{-\lambda\, f_l(X_i)}}{\sum_{q\in\mathcal{Y}} e^{-\lambda\, f_q(X_i)}}\, \mathbb{1}_{\left\{l=\operatorname{argmin}_{q\in\mathcal{Y}} f_q(X_i)\right\}} \;=\; 1,$$

which proves the convergence in probability of $\delta_\pi^\lambda(X_i)$ to $\delta_\pi^B(X_i)$ as λ goes to infinity for all $\pi \in \Pi$.

□

Proof of the pointwise convergence of V_λ to V_B over the entire simplex. Let $\pi \in \mathbb{S}$ fixed.

- If $\pi \in \Pi \subset \mathbb{S}$, then from the definition of Π, for all $t \in \mathcal{T}$, there exists a unique $l \in \mathcal{Y}$ such that $l = \operatorname{argmin}_{q\in\mathcal{Y}} \sum_{j\in\mathcal{Y}} L_{jq}\, \pi_j\, \hat{p}_{jt}$. Hence, from Lemma 3, it follows that for all $t \in \mathcal{T}$,

$$\frac{e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jl}\, \hat{p}_{jt}\, \pi_j}}{\sum_{q=1}^{K} e^{-\lambda \sum_{j\in\mathcal{Y}} L_{jq}\, \hat{p}_{jt}\, \pi_j}} \xrightarrow[\lambda\to+\infty]{} \mathbb{1}_{\left\{\sum_{j\in\mathcal{Y}} L_{jl}\, \pi_j\, \hat{p}_{jt} = \min_{q\in\mathcal{Y}} \sum_{j\in\mathcal{Y}} L_{jq}\, \pi_j\, \hat{p}_{jt}\right\}}. \tag{41}$$

  Applying this in equations (12) and (13), we finally get

$$V_\lambda(\pi) \xrightarrow[\lambda\to+\infty]{} \sum_{k\in\mathcal{Y}} \sum_{t\in\mathcal{T}} \sum_{l\in\mathcal{Y}} \pi_k\, L_{kl}\, \hat{p}_{kt}\, \mathbb{1}_{\left\{\sum_{j\in\mathcal{Y}} L_{jl}\, \pi_j\, \hat{p}_{jt} = \min_{q\in\hat{\mathcal{Y}}} \sum_{j\in\mathcal{Y}} L_{jq}\, \pi_j\, \hat{p}_{jt}\right\}} \tag{42}$$

  which corresponds to the discrete Bayes risk $V_B(\pi)$ introduced in equation (9).

- If $\pi \in \mathbb{S}\backslash\Pi$, there exists a sequence $\left(\pi^{(n)}\right)_{n\in\mathbb{N}} \in \Pi$ such that $\lim_{n\to\infty} \pi^{(n)} = \pi$. Since the function V_λ is continuous over the simplex $\mathbb{S}$, it follows that $\lim_{n\to\infty} V_\lambda\left(\pi^{(n)}\right) = V_\lambda(\pi)$, and thus

$$\lim_{n\to\infty} \lim_{\lambda\to\infty} V_\lambda\left(\pi^{(n)}\right) \;=\; \lim_{\lambda\to\infty} \lim_{n\to\infty} V_\lambda\left(\pi^{(n)}\right) \;=\; \lim_{\lambda\to\infty} V_\lambda(\pi). \tag{43}$$

  Moreover, since $\pi^{(n)} \in \Pi$ for all $n \in \mathbb{N}$, then from the previous item we have

$$\forall n \in \mathbb{N},\; \lim_{\lambda\to\infty} V_\lambda\left(\pi^{(n)}\right) \;=\; V_B\left(\pi^{(n)}\right).$$

  Furthermore, since the function V_B is continuous over the simplex $\mathbb{S}$, it follows that

$$\lim_{n\to\infty} \lim_{\lambda\to\infty} V_\lambda\left(\pi^{(n)}\right) \;=\; \lim_{n\to\infty} V_B\left(\pi^{(n)}\right) \;=\; V_B\left(\lim_{n\to\infty} \pi^{(n)}\right) \;=\; V_B(\pi). \tag{44}$$

  Hence, from equations (43) and (44), we obtain that $\lim_{\lambda\to\infty} V_\lambda(\pi) = V_B(\pi)$. This property holds for all $\pi \in \mathbb{S}\backslash\Pi$.

Finally, from the two previous items, for all $\pi \in \mathbb{S}$, $V_\lambda(\pi)$ converges pointwise to the Bayes risk $V_B(\pi)$ as the temperature parameter λ goes to infinity. □

B.3 Detailed proof of Corollary 1

Let $\bar{\pi} = \operatorname{argmax}_{\pi \in \mathbb{S}} V_B(\pi)$ and let us consider that the discrete minimax criterion $\delta^B_{\bar{\pi}}$ (Gilet et al., 2020) is not an equalizer classifier. Moreover, let us remind that

$$\Lambda(\bar{\pi}) \;=\; \left\{ \lambda > 0 : V_B(\bar{\pi}) \leq \max_{\pi \in \mathbb{S}} V_\lambda(\pi) \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right) \right\}, \tag{45}$$

and that for all $\pi \in \mathbb{S}$, $V_B(\pi) = \inf_{\lambda > 0} V_\lambda(\pi)$. As a consequence of Proposition 2, we have

$$\forall \varepsilon > 0,\ \forall \pi \in \mathbb{S},\ \exists\, \eta_{\varepsilon,\pi} > 0,\ \forall \lambda > \eta_{\varepsilon,\pi} \ :\ V_\lambda(\pi) - V_B(\pi) < \varepsilon. \tag{46}$$

In particular, let us consider $\varepsilon = \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right) - V_B(\bar{\pi})$. It follows that

$$\forall \pi \in \mathbb{S},\ \exists\, \eta_{\varepsilon,\pi} > 0,\ \forall \lambda > \eta_{\varepsilon,\pi} \ :\ V_\lambda(\pi) \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right).$$

Given $\eta > 0$, let us consider $\mathcal{U}_\eta = \left\{ \pi \in \mathbb{S} : \forall \lambda > \eta, V_\lambda(\pi) \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right) \right\}$ and its complementary $\mathcal{U}^c_\eta = \left\{ \pi \in \mathbb{S} \setminus \mathcal{U}_\eta \right\}$. Now, if exists $\pi' \in \mathcal{U}^c_\eta$, then it follows from equation (46) that

$$\exists\, \eta' > \eta,\ \forall \lambda > \eta' \ :\ \begin{cases} V_\lambda(\pi') \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right) \\ V_\lambda(\pi) \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right), \quad \forall \pi \in \mathcal{U}_\eta. \end{cases}$$

Hence, $\mathcal{U}_{\eta'} = \mathcal{U}_\eta \cup \left\{ \pi \in \mathcal{U}^c_\eta : \forall \lambda > \eta', V_\lambda(\pi) \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right) \right\}$. From equation (46), we can apply this reasoning until that there exists $\eta_\star > 0$ such that $\mathcal{U}_{\eta_\star} = \mathbb{S}$, which implies that

$$\forall \pi \in \mathbb{S},\ \forall \lambda > \eta_\star \ :\ V_\lambda(\pi) \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right).$$

Hence, it follows that

$$\forall \lambda > \eta_\star \ :\ \max_{\pi \in \mathbb{S}} V_\lambda(\pi) \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right). \tag{47}$$

Moreover, let us remind that for all $\lambda > 0$, $V_B(\bar{\pi}) \leq \max_{\pi \in \mathbb{S}} V_\lambda(\pi)$ since V_B corresponds to the Bayes risk. When considering this in equation (47), it follows that

$$\forall \lambda > \eta_\star \ :\ V_B(\bar{\pi}) \leq \max_{\pi \in \mathbb{S}} V_\lambda(\pi) \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right). \tag{48}$$

Hence, the set $\Lambda(\bar{\pi})$ defined in equation (14) is equal to $\Lambda(\bar{\pi}) \;=\; \{\lambda > \eta_\star\}$. This proves that $\Lambda(\bar{\pi})$ is not empty.

Now, let $\lambda \in \Lambda(\bar{\pi})$, and let us remind that

$$\mathcal{B}_\lambda(\bar{\pi}) \;=\; \left\{ \pi \in \mathbb{S} : V_B(\bar{\pi}) \leq V_\lambda(\pi) \leq \max_{k \in \mathcal{Y}} \hat{R}_k \left(\delta^B_{\bar{\pi}} \right) \right\}, \tag{49}$$

and let us define $\pi^\star = \operatorname{argmax}_{\pi \in \mathbb{S}} V_\lambda(\pi)$. From equation (48), $\pi^\star \in \mathcal{B}_\lambda(\bar{\pi})$, which proves that $\mathcal{B}_\lambda(\bar{\pi})$ is not empty. Another candidate belonging to $\mathcal{B}_\lambda(\bar{\pi})$ is $\bar{\pi}$. Indeed, since $V_B(\bar{\pi}) = \inf_{\lambda > 0} V_\lambda(\bar{\pi})$, and since $V_\lambda(\bar{\pi}) \leq \max_{\pi \in \mathbb{S}} V_\lambda(\pi)$, it follows from equation (48) that $\bar{\pi} \in \mathcal{B}_\lambda(\bar{\pi})$. □

B.4 Detailed proof of Lemma 1

Let $\lambda \in \Lambda(\bar{\pi})$ fixed. We first prove that the softmin randomized decision rule $\delta^{\lambda}_{\pi^\star}$ associated with the priors $\pi^\star \in \mathbb{S}$ is an equalizer classifier if and only if $G(\pi^\star) = 0$. Then, we prove the existence of such a root $\pi^\star$ in $\mathcal{B}_\lambda(\bar{\pi})$.

Equivalence between G-root and equalizer classifier. Let $\pi^\star \in \mathbb{S}$. Let us remind that $\delta^{\lambda}_{\pi^\star}$ is an equalizer classifier if and only if

$$\hat{R}_1\left(\delta^{\lambda}_{\pi^\star}\right) \;=\; \cdots \;=\; \hat{R}_K\left(\delta^{\lambda}_{\pi^\star}\right). \tag{50}$$

According to equations (12) and (50), this is equivalent to say that

$$\forall k \in \mathcal{Y},\; \hat{R}_k\left(\delta^{\lambda}_{\pi^\star}\right) = V_\lambda(\pi^\star) \quad \Leftrightarrow \quad \forall k \in \mathcal{Y},\; \hat{R}_k\left(\delta^{\lambda}_{\pi^\star}\right) - V_\lambda(\pi^\star) = 0 \quad \Leftrightarrow \quad G(\pi^\star) = 0.$$

In other words, $\delta^{\lambda}_{\pi^\star}$ is an equalizer classifier if and only if $G(\pi^\star) = 0$. □

Existence of a G-root $\pi^\star$ in $\mathcal{B}_\lambda(\bar{\pi})$. We first prove that there exists a G-root $\pi^\star$ in the simplex $\mathbb{S}$ and then that this root $\pi^\star$ necessarily belongs to $\mathcal{B}_\lambda(\bar{\pi}) \subset \mathbb{S}$.

- Let us remind that for all $k \in \mathcal{Y}$, the functions $g_k : \mathbb{S} \to \mathbb{R}$ defined in equation (16) are given by $g_k(\pi) = \hat{R}_k\left(\delta^{\lambda}_{\pi}\right) - V_\lambda(\pi)$. Under Assumption 2,
  - $\forall k \in \mathcal{Y}, \exists\, \varepsilon_k > 0, \forall \pi \in \mathcal{Q}_k = \{\pi \in \mathbb{S} : \pi_k < \varepsilon_k\} : \hat{R}_k\left(\delta^{\lambda}_{\pi}\right) \geq V_\lambda(\pi) \Rightarrow g_k(\pi) \geq 0.$
  - $\forall k \in \mathcal{Y}, \exists\, \eta_k > 0, \forall \pi \in \mathcal{U}_k = \{\pi \in \mathbb{S} : \pi_k > \eta_k\} : \hat{R}_k\left(\delta^{\lambda}_{\pi}\right) \leq V_\lambda(\pi) \Rightarrow g_k(\pi) \leq 0.$

  Moreover, since the functions $g_k : \mathbb{S} \to \mathbb{R}$ are all continuous over the simplex $\mathbb{S}$ and similarly to the Poincaré–Miranda Theorem (Kulpa, 1997; Frankowska, 2018; Fonda and Gidoni, 2016; Mawhin, 2007), there exists a root $\pi^\star \in \mathbb{S}$ such that for all $k \in \mathcal{Y}$, $g_k(\pi^\star) = 0$, and thus $G(\pi^\star) = 0$.
- We now need to ensure that this root $\pi^\star$ necessarily belongs to $\mathcal{B}_\lambda(\bar{\pi}) \subset \mathbb{S}$. Let us remind the definition (15) of $\mathcal{B}_\lambda(\bar{\pi})$:

  $$\mathcal{B}_\lambda(\bar{\pi}) \;=\; \left\{\pi \in \mathbb{S} \;:\; V_B(\bar{\pi}) \;\leq\; V_\lambda(\pi) \;\leq\; \max_{k \in \mathcal{Y}} \hat{R}_k\left(\delta^{B}_{\bar{\pi}}\right)\right\}.$$

  Since $\lambda \in \Lambda(\bar{\pi})$, it follows from equation (14) that

  $$V_\lambda(\pi^\star) \;\leq\; \max_{\pi \in \mathbb{S}} V_\lambda(\pi) \;\leq\; \max_{k \in \mathcal{Y}} \hat{R}_k\left(\delta^{B}_{\bar{\pi}}\right). \tag{51}$$

  Moreover, since $G(\pi^\star) = 0$, the softmin randomized decision rule $\delta^{\lambda}_{\pi^\star}$ associated with the priors $\pi^\star$ is an equalizer classifier. Hence, from equations (5) and (50), it follows that the average risk associated with the prior probability shift $\bar{\pi}$ is

  $$\hat{r}\left(\bar{\pi}, \delta^{\lambda}_{\pi^\star}\right) \;=\; V_\lambda(\pi^\star).$$

  Moreover, since $V_B(\bar{\pi}) = \min_{\delta \in \Delta} \hat{r}\left(\bar{\pi}, \delta\right)$, it follows that

  $$V_B(\bar{\pi}) \;\leq\; V_\lambda(\pi^\star). \tag{52}$$

  Finally, from the bounds (51) and (52), the root $\pi^\star$ necessarily belongs to $\mathcal{B}_\lambda(\bar{\pi})$.

This concludes the proof. □

C Non-convexity of the optimization problem (18)

This section illustrates the non-convexity of the function $\|G(\pi)\|_2^2$ over the simplex $\mathbb{S}$ on the synthetic dataset presented in Figure 2. We can observe in Figure 7 that the non-convexity of $\|G(\pi)\|_2^2$ is accentuated when the temperature parameter λ is high. This is indeed a consequence of Proposition 2: the more λ increases, the more V_λ converges to V_B (which is piecewise affine over the simplex), and thus the more the class-conditional risks of the softmin discrete minimax classifier converge to the risks per class of the discrete minimax classifier (which are a subgradient of V_B as shown in (Gilet et al., 2020)). To conclude this part, let us note that several experiments conduct us to presume the following conjecture.

Conjecture 1. *The function $\mathcal{G} : \pi \mapsto \|G(\pi)\|_2^2$ is strictly quasi-convex over the simplex $\mathbb{S}$.*

The mathematical proof of Conjecture 1 is not straightforward for a general context $K > 2$ classes and is still under investigation. This property is not necessary for solving (18) as discussed in the main paper but it could be convenient for optimizing the convergence of gradient based algorithms toward the priors $\pi^\star$ solution of (18).

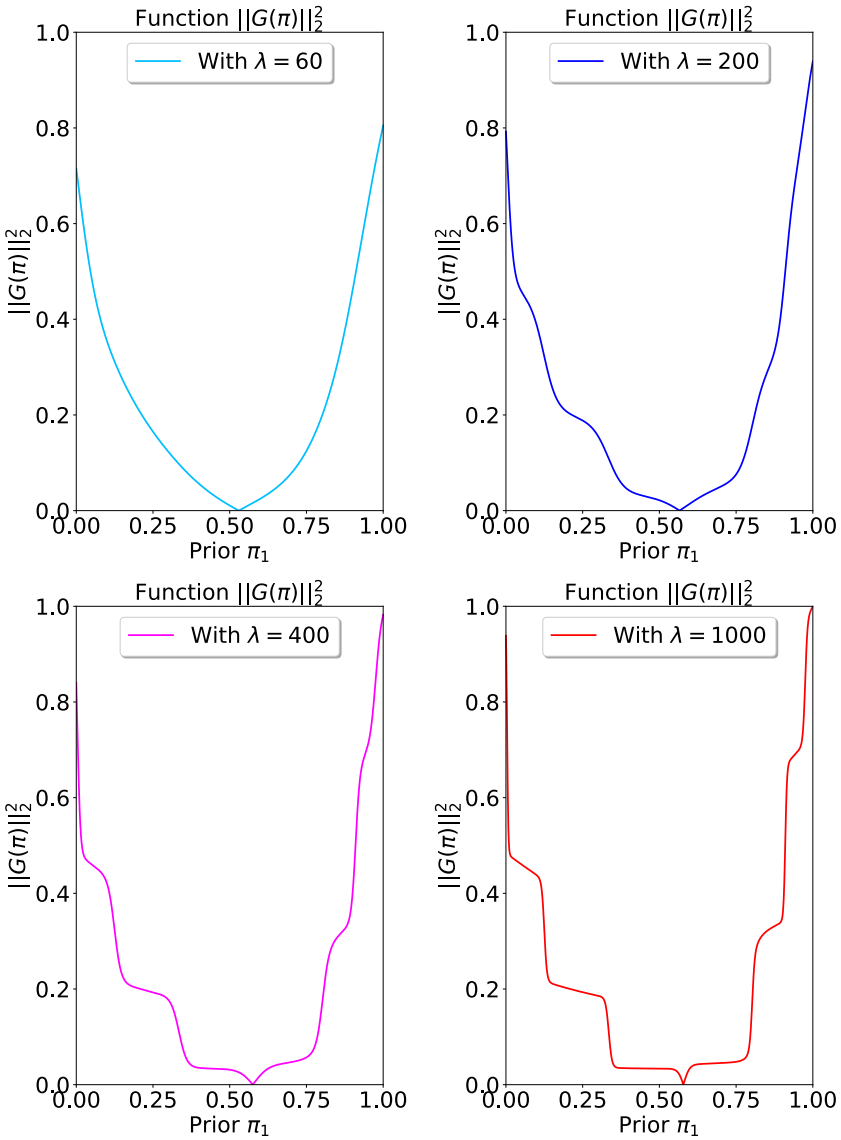

Figure 7: Non-convexity of the function $\|G(\pi)\|_2^2$ over the simplex $\mathbb{S}$ with respect to the parameter λ.

D Softmin Discrete Minimax Algorithm

As illustrated in Supplementary Material C, the minimization problem (18) is not necessary convex. Keeping aside the presumed quasi-convexity of this problem, we first consider well studied methods which are designed to solve non-convex optimization problems in a general context. To this aim, we mentioned previously that several approaches such that gradient based algorithms or Monte-Carlo based algorithms can be relevant for solving (18). When the number of classes K is small enough, Monte-Carlo based algorithms using the Dirichlet distribution (as summarized in Subsection D.1) are able to converge efficiently to the global solution. However, the complexity of this kind of method becomes too high when the number of classes becomes too large. In such a complex context, stochastic projected gradient based algorithm appear more convenient.

While the previously mentioned approaches allow to address the optimization problem (18) and thus to theoretically reach the objectives of the present paper, we present in Subsection D.2 a more convenient algorithm which is conjectured to converge to the priors $\pi^\star$ solution of (18) and which showed efficiency on several experiments, especially on the famous CIFAR-100 database.

For the following, let us denote $\mathcal{G} : \pi \to \mathbb{S}$ such that

$$\mathcal{G}(\pi) = \|G(\pi)\|_2^2. \tag{53}$$

Finally, let us note that for all these previously mentioned algorithms, our approach does not need to resample the training set at each iteration n. Indeed, the priors $\pi^{(n)}$ and $\pi^\star$ are used only analytically, which enables us to consider all the information provided in the training set for computing our softmin discrete minimax classifier $\delta_{\pi^\star}^{\lambda}$.

Remark 1. *Let us note that the temperature parameter $\lambda \in \Lambda(\bar{\pi})$ is beforehand set. In practice, an efficient way to set it is to consider the priors $\bar{\pi}$ and to increase λ until that $V_\lambda(\bar{\pi}) < \max_{k\in\mathcal{Y}} \hat{R}_k(\delta_{\bar{\pi}}^B)$.*

D.1 Monte-Carlo based algorithm

The procedure for computing our softmin discrete minimax classifier $\delta_{\pi^\star}^{\lambda}$ using a Monte-Carlo based approach is summarized in the step by step Algorithm 1. In Algorithm 1, N denotes the maximum number of iterations. Moreover, Lemma 1 implies that $\min_{\pi\in\mathbb{S}} \mathcal{G}(\pi) = 0$ and we therefore consider a threshold $\varepsilon > 0$ such that we accept a sample point $\pi^{(n)} \in \mathbb{S}$ as a solution if $\mathcal{G}(\pi^{(n)}) < \varepsilon$. In practice, this algorithm is especially convenient, efficient and fast when dealing with a small number of classes (for instance when $K \leq 5$). Indeed, our constraint set is the simplex $\mathbb{S}$, which is compact and for which it is easy to uniformly sample points using the famous Dirichlet distribution. However, this algorithm is not relevant when the number of classes K is large since the computation time would become too huge.

Algorithm 1 Softmin Discrete Minimax Classifier (using a Monte-Carlo based approach)

Input: Training set $\mathcal{S} = \{(Y_i, X_i), i \in \mathcal{I}\}$, K, L, N, λ, $\varepsilon > 0$.
Compute the $\hat{p}_{kt}$ values as in equation (6)
Initialize $\pi^{(1)} = [1/K, \ldots, 1/K]$
Initialize $\pi^\star = \pi^{(1)}$
Initialize $\mathcal{G}^\star = \mathcal{G}\left(\pi^{(1)}\right)$
for $n = 1$ **to** N **do**
  Sample uniformly a point $\pi^{(n+1)} \in \mathbb{S}$ using the Dirichlet distribution
  if $\mathcal{G}\left(\pi^{(n+1)}\right) < \mathcal{G}^\star$ **then**
    $\mathcal{G}^\star \leftarrow \mathcal{G}\left(\pi^{(n+1)}\right)$
    $\pi^\star \leftarrow \pi^{(n+1)}$
    if $\mathcal{G}^\star \leq \varepsilon$ **then**
      Break
    end if
  end if
end for
Output: Priors $\pi^\star$ solution of (18) and $\delta_{\pi^\star}^{\lambda}$ provided by (19).

D.2 Projected descent based algorithm

Another relevant approach for computing the priors $\pi^\star$ solution of the minimization problem (18) would be to consider a projected gradient algorithm following the scheme

$$\pi^{(n+1)} = \mathrm{P}_{\mathbb{S}}\left(\pi^{(n)} - \frac{\gamma_n}{\eta_n}\nabla\mathcal{G}\left(\pi^{(n)}\right)\right), \tag{54}$$

where at each iteration $n \geq 1$, $\nabla\mathcal{G}\left(\pi^{(n)}\right)$ is the gradient of $\mathcal{G}$ at the point $\pi^{(n)}$, γ_n denotes the gradient step, $\eta_n = \max\{1, \|\nabla\mathcal{G}\left(\pi^{(n)}\right)\|_2\}$, and where $\mathrm{P}_{\mathbb{S}}$ denotes the exact projection onto the simplex $\mathbb{S}$. However, the gradient of $\mathcal{G}$ is complex to compute at each iteration, especially in a high dimensional simplex $\mathbb{S}$. In other words, although this approach is theoretically proved to achieve satisfying results, it requires a huge computation time when dealing with a large number of classes. In this context, stochastic gradient based methods can appear relevant and can be implemented by the users. In this section, we propose to discuss a faster and more convenient descent algorithm which is conjectured to converge toward the priors $\pi^\star$ solution of (18) based on the following conjecture.

Conjecture 2. *Given $\pi \in \mathbb{S}$ and when considering $G(\pi) = \left[g_1(\pi), \ldots, g_K(\pi)\right]$ as defined in equation (16), the vector $-G(\pi)$ is a descent direction of the function $\mathcal{G}(\pi)$.*

Figure 8 illustrates this conjecture. We can observe that if $g_k(\pi^{(n)}) = \hat{R}_k\left(\delta^\lambda_{\pi^{(n)}}\right) - V_\lambda(\pi^{(n)}) > 0$, then $\pi_k^{(n+1)} = \pi_k^{(n)} + \frac{\gamma_n}{\eta_n} g_k\left(\pi^{(n)}\right)$ is a descent direction with respect to $\mathcal{G}$. Moreover, if $g_k(\pi^{(n)}) < 0$, then $\pi_k^{(n+1)} = \pi_k^{(n)} + \frac{\gamma_n}{\eta_n} g_k\left(\pi^{(n)}\right)$ is a descent direction too.

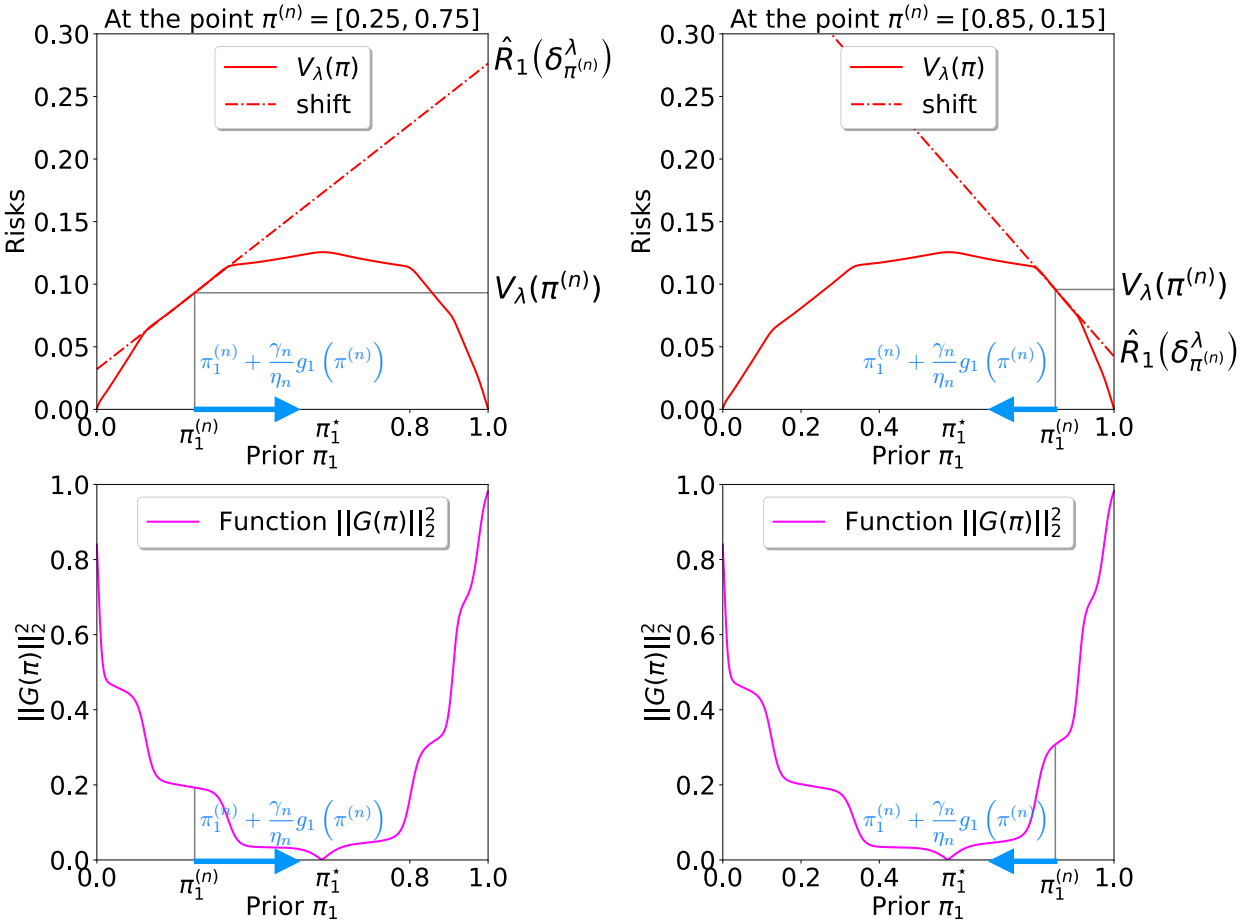

Figure 8: Illustration of the descent iteration algorithm (55) for $K = 2$ classes.

We therefore propose to consider the following projected descent algorithm

$$\pi^{(n+1)} = \mathrm{P}_{\mathbb{S}}\left(\pi^{(n)} + \frac{\gamma_n}{\eta_n} G\left(\pi^{(n)}\right)\right), \tag{55}$$

where, at each iteration $n \geq 1$, γ_n denotes the descent step, $\eta_n = \max\{1, \|G\left(\pi^{(n)}\right)\|_2\}$, and $\mathrm{P}_{\mathbb{S}}$ denotes the exact projection onto the simplex $\mathbb{S}$. The following conjecture studies the convergence of the iterates (55) toward the priors $\pi^\star$ solution of (18).

Conjecture 3. *Under Conjecture 1 and Conjecture 2 and when considering any sequence of steps $(\gamma_n)_{n\geq 1}$ satisfying*

$$\inf_{n\geq 1} \gamma_n > 0, \quad \sum_{n=1}^{+\infty} \gamma_n^2 < +\infty, \quad \sum_{n=1}^{+\infty} \gamma_n = +\infty, \tag{56}$$

the sequence of iterates (55) converges to a solution $\pi^\star$ of (18), whatever the initialization $\pi^{(1)} \in \mathbb{S}$.

The procedure for computing the softmin discrete minimax classifier $\delta^{\lambda}_{\pi^\star}$ using our new projected descent based scheme is summarized in the step by step Algorithm 2. In Algorithm 2, N denotes the maximum number of iterations. Moreover, since $\min_{\pi\in\mathbb{S}} \mathcal{G}(\pi) = 0$, we consider the threshold $\varepsilon > 0$ such that we accept a point $\pi^{(n)} \in \mathbb{S}$ as a solution if $\mathcal{G}(\pi^{(n)}) < \varepsilon$. This algorithm showed efficiency on several experiments, especially on the famous CIFAR-100 database (see Section 4).

Algorithm 2 Softmin Discrete Minimax Classifier (using projected descent based iterations)

Input: Training set $\mathcal{S} = \left\{(Y_i, X_i), i \in \mathcal{I}\right\}$, K, L, N, $\lambda \in \Lambda(\bar{\pi})$, $\varepsilon > 0$.
Compute the $\hat{p}_{kt}$ values as in equation (6)
Initialize $\pi^{(1)} = \bar{\pi}$
Initialize $\pi^\star = \pi^{(1)}$
Initialize $\mathcal{G}^\star = \mathcal{G}\left(\pi^{(1)}\right)$
for $n = 1$ **to** N **do**
  Compute $G\left(\pi^{(n)}\right)$ using equation (16)
  Compute $\mathcal{G}\left(\pi^{(n)}\right) = \|G\left(\pi^{(n)}\right)\|_2^2$
  if $\mathcal{G}\left(\pi^{(n)}\right) < \mathcal{G}^\star$ **then**
    $\mathcal{G}^\star \leftarrow \mathcal{G}\left(\pi^{(n)}\right)$
    $\pi^\star \leftarrow \pi^{(n)}$
    if $\mathcal{G}^\star \leq \varepsilon$ **then**
      Break
    end if
  end if
  $\gamma_n \leftarrow \xi(n)$, where the sequence $(\xi(n))_{n\in\mathbb{N}}$ satisfies condition (56)
  $\eta_n \leftarrow \max\left\{1, \|G\left(\pi^{(n)}\right)\|_2\right\}$
  $\pi^{(n+1)} = \mathrm{P}_{\mathbb{S}}\left(\pi^{(n)} + \frac{\gamma_n}{\eta_n} G\left(\pi^{(n)}\right)\right)$
end for
Output: Priors $\pi^\star$ solution of (18) and $\delta^{\lambda}_{\pi^\star}$ provided by (19).

E Additional experiment information, results and settings

E.1 Settings for the experiments on the CIFAR-100 database

Database settings. We consider the famous CIFAR-100 database (Krizhevsky, 2009) which contains 60,000 images with $K = 100$ classes and for which the class proportions are perfectly balanced. The scope of this experiment is not to compare our method with other state of the arts classifiers in minimizing the average risk (1) but to illustrate Theorem 1 when the number of classes K is large. While the initial database was splitted into a training set and a test set containing respectively 50,000 and 10,000 images, we decided to shuffle the 60,000 images in order to compose a new pair of training and test sets, each one containing 30,000 instances. Both the training and test sets satisfied the balanced class proportions $\hat{\pi} = [1/100, \ldots, 1/100]$ as in the initial datasets. These settings were considered in order to have a larger number of test images in each class and to avoid possible covariate shifts between the initial training and test sets (and thus to satisfy Assumption 1).

CNN and hyper-parameters settings. For this experiment we considered the deep features extracted from the last hidden layer of the convolutional neural networks EfficientNet-B0 (Tan and Le, 2019). We then discretized the deep features using the Kmeans procedure with $T = 1200$ centroids (each centroid corresponding to a discrete profile). We compared on the same discretized deep feature space the Discrete Bayes Classifier $\delta^{\mathrm{B}}_{\hat{\pi}}$ [given by (7) when considering the priors $\hat{\pi}$], the Discrete Minimax Classifier $\delta^{\mathrm{B}}_{\bar{\pi}}$ [given by (7) when considering the priors $\bar{\pi} = \operatorname{argmax}_{\pi \in \mathbb{S}} V_B(\pi)$] and our Softmin Discrete Minimax Classifier $\delta^{\lambda}_{\pi^\star}$. We chose to compare these three algorithms in order to illustrate the impact of both the priors $\hat{\pi}, \bar{\pi}, \pi^\star \in \mathbb{S}$ and the randomization on the same deep feature space partitioning. Regarding our Softmin Discrete Minimax Classifier $\delta^{\lambda}_{\pi^\star}$, we computed the priors $\pi^\star$ solution of (18) using Algorithm 2.

Additional results. Another interesting experiment is to decrease the number of centroids T when partitioning the deep feature space. This number of discrete profiles T has indeed an impact on the results, especially since the smaller T is, the more sharpened the Bayes risk V_B becomes. We therefore propose here a new experiment when considering $T = 800$ discrete profiles. Figure 9 displays these additional results. Although the CNN-DMC algorithm converged when computing the least favorable priors $\bar{\pi} = \operatorname{argmax}_{\pi \in \mathbb{S}} V_B(\pi)$, its class-conditional risks appear more imbalanced for $T = 800$ than for $T = 1200$ since the Bayes risk V_B is more sharpened with this smaller number of discrete profiles. Here again, we can observe that our new Sotmin-DMC criterion gets the best results in the difficult task of balancing the class-conditional risks and achieved $\max_{k \in \mathcal{Y}} \hat{R}_k(\delta^{\lambda}_{\pi^\star}) \leq \max_{k \in \mathcal{Y}} \hat{R}_k(\delta^{B}_{\bar{\pi}})$, which especially illustrates Theorem 1.

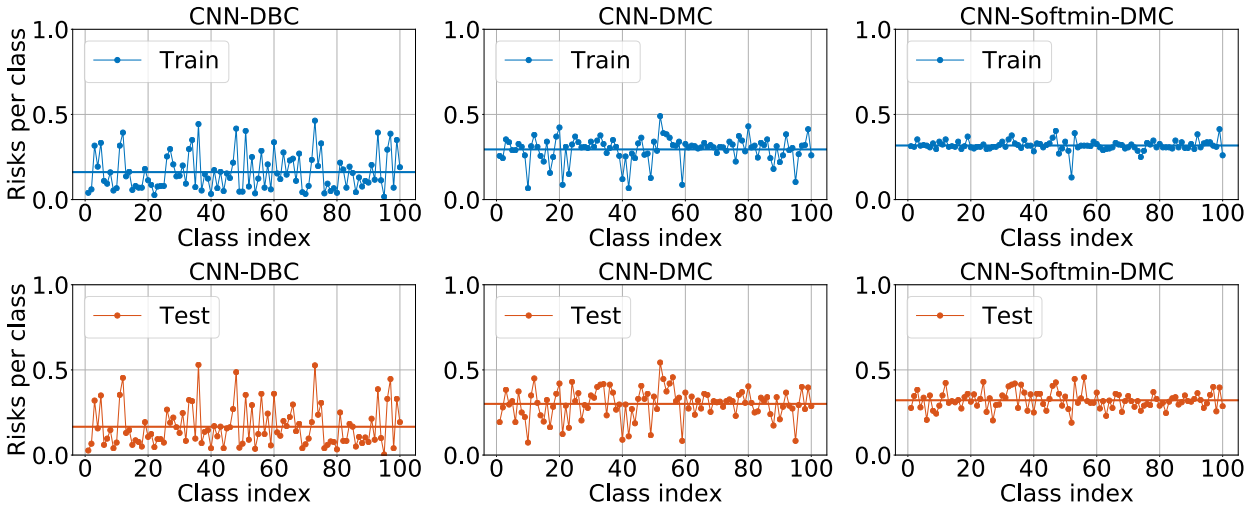

Figure 9: CIFAR-100 database: Class-conditional risks associated with the CNN-DBC, CNN-DMC and CNN-Softmin-DMC classifiers on both the training and test datasets when considering $T = 800$ discrete profiles.

E.2 Databases descriptions

Diabete Database (Johannes, 1988). The objective of this database is to predict the onset of diabetes based on diagnostic measurements and was originally studied by the National Institute of Diabetes and Digestive and Kidney Diseases. This database contains the measurements of 8 clinical and biological features (*Number of times pregnant,*

Plasma glucose concentration, Diastolic blood pressure, Triceps skin fold thickness, 2-Hour serum insulin, BMI, Diabetes pedigree function, Age) for 768 patients. We have $K = 2$ classes, where class 2 corresponds to the patients who tested positive for diabetes. The class proportions of this dataset are $\hat{\pi} = [0.65, 0.35]$. For this database, we considered the $L_{0\text{-}1}$ loss function.

APS Failure Trucks database (Scania, 2016). This real condition monitoring database focuses on the Air Pressure System (APS) used for various functions in Scania trucks such as braking and gear changes. Measurements of a specific APS component were collected from heavy Scania trucks in everyday use. The goal is to predict a potential failure of this component. We therefore have $K = 2$ classes where the class 1 corresponds to the APS without failures and class 2 to the defect APS components. For this database, the costs of classification errors were provided by the experts of the application domain:

$$L = \begin{bmatrix} 0 & 10 \\ 500 & 0 \end{bmatrix}, \tag{57}$$

so that the cost of predicting a nonexistent failure is $\$10$, while the cost of missing a failure is $\$500$. After removing missing values, the database contains the measurements of 69,309 samples, of which 68,494 do not present any failure and 815 do present a failure. Hence, the class proportions $\hat{\pi} = [0.9882, 0.0118]$ are highly imbalanced, which highly complicates the task of predicting a failure. Finally, each instance is described by $d = 130$ numeric features.

MedMNIST databases (Yang et al., 2020; Yang et al., 2020). We consider here three real medical databases (*OCTMNIST, DermaMNIST, BreastMNIST*) which differ in the number of images, the number of classes and the class proportions. 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 (Yang et al., 2020) Each database contains a training set, a validation set and a test set with 28×28 images. Table 3 provides an overview of each database and Figure 10 highlights their associated class proportions. Let us note that the OCTMNIST database presents prior probability shifts between the validation and the test sets. Finally, we considered the famous $L_{0\text{-}1}$ loss function for all these databases.

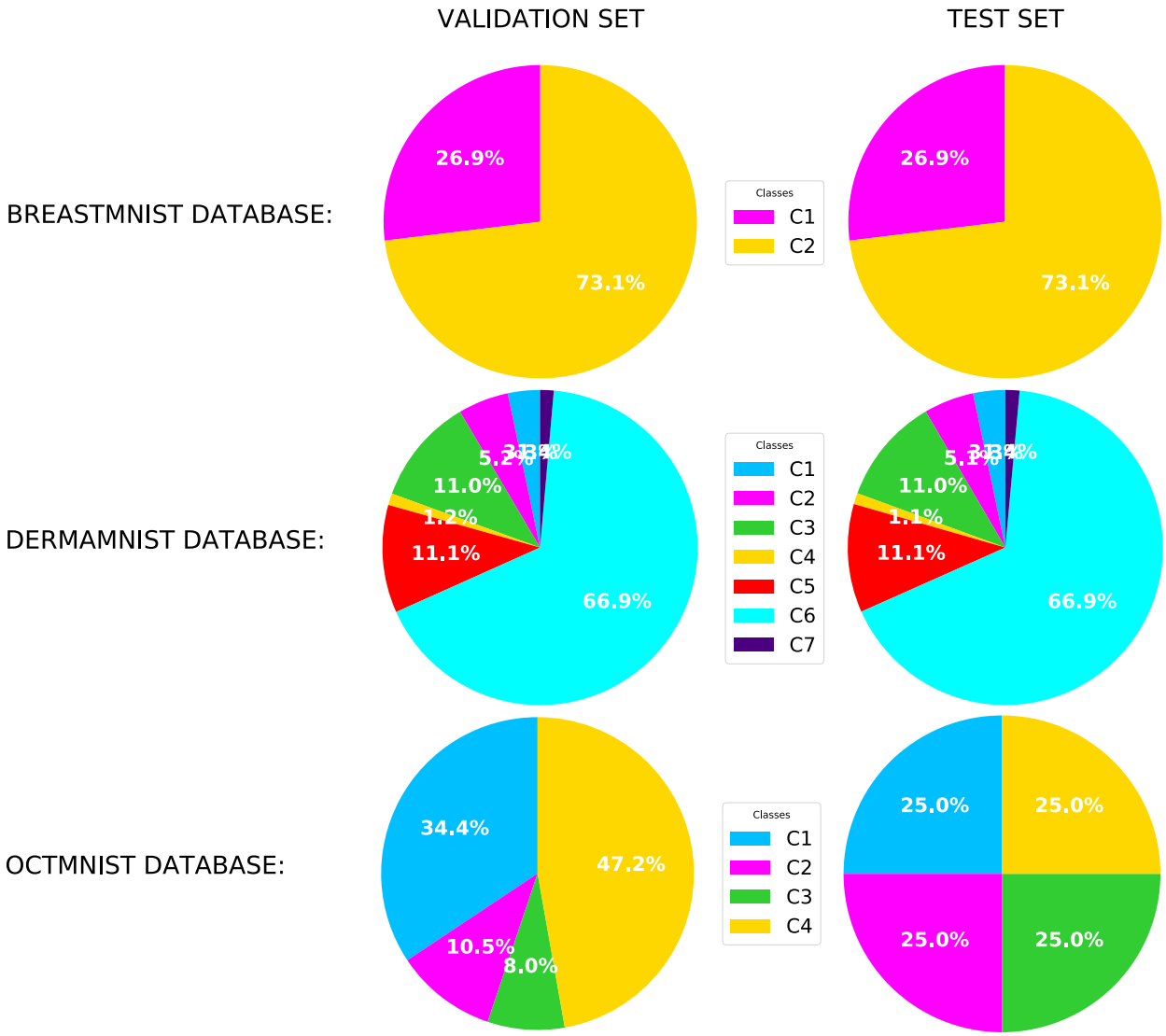

Figure 10: Class proportions of the Validation and Test sets for each MedMNIST database.