Evidential multinomial logistic regression for multiclass classifier calibration

Philippe Xu Franck Davoine Thierry Denœux

Sorbonne universités
Université de technologie de Compiègne
CNRS, Heudiasyc UMR 7253
CS 60 319, 60 203 Compiègne cedex, France
Email: philippe.xu@utc.fr

***Abstract*—The calibration of classifiers is an important task in information fusion. To compare or combine the outputs of several classifiers, they need to be represented in a common space. Probabilistic calibration methods transform the output of a classifier into a posterior probability distribution. In this paper, we introduce an evidential calibration method for multiclass classification problems. Our approach uses an extension of multinomial logistic regression to the theory of belief functions. We demonstrate that the use of belief functions instead of probability distributions is often beneficial. In particular, when different classifiers are trained with unbalanced amount of training data, the gain achieved by our evidential approach can become significant. We applied our method to the calibration of multiclass SVM classifiers which were constructed through a "one-vs-all" framework. Experiments were conducted using six different datasets from the UCI repository.**

I. Introduction

Combining the outputs of multiple classifiers is an important and challenging task in machine learning. In particular, it becomes difficult when these classifiers return some pieces of information that are not directly comparable. Calibration consists in transforming the output of a classifier into a unique representation. Calibration is usually considered within a probabilistic framework by converting the output of a classifier into a posterior probability distribution. Many existing methods are designed to calibrate binary classifiers [1]–[3]. In particular, the logistic regression-based method proposed by Platt [1] is among the most commonly used ones. Recently, an extension of this method to the evidential framework was proposed [4]. It was shown that the use of belief functions instead probability distributions can better model the uncertainty of the calibration process.

In a multiclass classification context, calibration is usually done through a binary decomposition framework. The classification problem is first converted into multiple binary subproblems. The "one-vs-one" [5] and "one-vs-all" [3] are the two mostly used strategies. Once all binary classifiers are calibrated, their outputs are combined into a multiclass probability distribution. Several studies [5], [6] analyzed the combination of pairwise calibrated classifiers within the "one-vs-one" framework. In this context, it has been shown that using more general representations such as belief functions [7], [8] or imprecise probabilities [9] gives better results compared to traditional probabilistic methods.

In this paper, we consider that a multiclass classifier returns a set of scores where each of them represents the support for one of the potential class label. Classifiers built from a "one-vs-all" framework naturally satisfy this constraint. The aim of the calibration is then to transform these scores into a probability distribution or belief function. More specifically, we will extend the probabilistic calibration method proposed by Milgram et al. [10] to the evidential framework.

The rest of the paper is organized as follows. In Section II, we provide an introduction to the theory of belief functions covering both prediction and statistical inference. Next, a probabilistic multiclass calibration method based on multinomial logistic regression is presented in Section III. This calibration method is then extended to the evidential framework in Section IV. Experimental results on the calibration of SVM classifiers are shown in Section V. Finally, Section VI concludes this paper.

II. Theory of belief functions

The theory of belief functions, also known as the Dempster-Shafer theory [11] or evidence theory, is a generalization of the theory of probability. It is closely linked to other theories such as random sets [12] or imprecise probabilities [13]. In this section, we introduce some basic notions of the theory of belief functions for both prediction and statistical inference.

A. Predictive belief functions

Let $\mathbf{x} \in \mathbb{X}$ be an observed instance of an object of unknown class $y \in \Omega$, where $\Omega = \{\omega_1, \ldots, \omega_K\}$ is called the frame of discernment. The knowledge about y induced by $\mathbf{x}$ can be represented by a mass function $m^{\Omega}_{\mathbf{x}} : 2^{\Omega} \rightarrow [0, 1]$ verifying

$$\sum_{A \subseteq \Omega} m^{\Omega}_{\mathbf{x}}(A) = 1, \qquad m^{\Omega}_{\mathbf{x}}(\emptyset) = 0. \tag{1}$$

The quantity $m^{\Omega}_{\mathbf{x}}(A)$, for a given subset $A \subseteq \Omega$, represents the belief committed exactly to the hypothesis $y \in A$. A set $A \subseteq \Omega$ such that $m^{\Omega}_{\mathbf{x}}(A) \neq 0$ is said to be a focal element of $m^{\Omega}_{\mathbf{x}}$.

Let m_1^Ω and m_2^Ω be two mass functions, generated by two independent pieces of evidence $\mathbf{x}_1$ and $\mathbf{x}_2$. They can be combined by Dempster's rule of combination yielding a new mass function $m_{1,2}^\Omega$ defined as

$$m_{1,2}^\Omega(\emptyset) = 0,\ m_{1,2}^\Omega(A) = \frac{1}{1-\kappa} \sum_{B \cap C = A} m_1^\Omega(B) m_2^\Omega(C), \quad (2)$$

where

$$\kappa = \sum_{B \cap C = \emptyset} m_1^\Omega(B) m_2^\Omega(C) \quad (3)$$

measures the amount of conflict between the two mass functions.

The information encoded by $m_{\mathbf{x}}^\Omega$ can also be equivalently represented by a belief function or a plausibility function, defined, respectively, as

$$Bel_{\mathbf{x}}^\Omega(A) = \sum_{B \subseteq A} m_{\mathbf{x}}^\Omega(B), \quad Pl_{\mathbf{x}}^\Omega(A) = \sum_{B \cap A \neq \emptyset} m_{\mathbf{x}}^\Omega(B), \quad (4)$$

for all $A \subseteq \Omega$. The degree of belief $Bel_{\mathbf{x}}^\Omega(A)$ represents the amount of evidence strictly supporting the hypothesis $y \in A$, while the plausibility $Pl_{\mathbf{x}}^\Omega(A) = 1 - Bel_{\mathbf{x}}^\Omega\left(\overline{A}\right)$ is the amount of evidence not contradicting it.

There exist a number of strategies to predict a class label from a mass function [14]. In this paper, we will use the optimistic decision rule which consists in selecting the singleton with maximum plausibility. The predicted label y^* is thus given by

$$y^* = \arg\max_{\omega \in \Omega} pl_{\mathbf{x}}^\Omega(\omega), \quad (5)$$

where $pl_{\mathbf{x}}^\Omega : \Omega \to [0,1]$ is the contour function associated with $m_{\mathbf{x}}^\Omega$ and defined as its plausibility on singletons

$$pl_{\mathbf{x}}^\Omega(\omega) = Pl_{\mathbf{x}}^\Omega(\{\omega\}), \quad \forall \omega \in \Omega. \quad (6)$$

The decision led by the optimistic rule given several contour functions can be computed in time linear in both the number of sources and the number of classes. Given two contour functions pl_1^Ω and pl_2^Ω, their combined contour function $pl_{1,2}^\Omega$ verifies

$$pl_{1,2}^\Omega(\omega) \propto pl_1^\Omega(\omega) pl_2^\Omega(\omega), \quad \forall \omega \in \Omega. \quad (7)$$

In contrast, computing the combination of two mass functions needs a time exponential in the number of classes. It can become computationally intractable when the number of classes or the number of mass functions to combine becomes large.

B. Statistical inference

For statistical inference, Shafer [11] proposed to construct a belief function from the likelihood function. Recently Denœux [15], [16] further justified this approach. Let $\mathbf{X} \in \mathbb{X}$ be some observable data that are generated from a density function $f_\theta(\mathbf{x})$ where $\theta \in \Theta$ is an unknown parameter. Given the outcome $\mathbf{x}$ of a random experiment, information about θ can be inferred. Shafer [11] proposed to build a belief function $Bel_{\mathbf{x}}^\Theta$ from the likelihood function. From the realization $\mathbf{X} = \mathbf{x}$, the likelihood function $L_{\mathbf{x}} : \theta \mapsto f_\theta(\mathbf{x})$ is normalized to yield the following contour function:

$$pl_{\mathbf{x}}^\Theta(\theta) = \frac{L_{\mathbf{x}}(\theta)}{\sup_{\theta' \in \Theta} L_{\mathbf{x}}(\theta')}, \quad \forall \theta \in \Theta. \quad (8)$$

The consonant plausibility function associated to this contour function is

$$Pl_{\mathbf{x}}^\Theta(A) = \sup_{\theta \in A} pl_{\mathbf{x}}^\Theta(\theta), \quad \forall A \subseteq \Omega. \quad (9)$$

The focal sets of $Bel_{\mathbf{x}}^\Theta$ are defined as

$$\Gamma_{\mathbf{x}}(\gamma) = \{\theta \in \Theta \mid pl_{\mathbf{x}}^\Theta(\theta) \geq \gamma\}, \quad \forall \gamma \in [0,1]. \quad (10)$$

The random sets formalism [12] is often used to represent belief functions over a continuous space. Given the Lebesgue measure λ on $[0,1]$ and the multi-valued mapping $\Gamma_{\mathbf{x}} : [0,1] \to 2^\Theta$, we have

$$Bel_{\mathbf{x}}^\Theta(A) = \lambda\left(\{\gamma \in [0,1] \mid \Gamma_{\mathbf{x}}(\gamma) \subseteq A\}\right), \quad (11a)$$

$$Pl_{\mathbf{x}}^\Theta(A) = \lambda\left(\{\gamma \in [0,1] \mid \Gamma_{\mathbf{x}}(\gamma) \cap A \neq \emptyset\}\right), \quad (11b)$$

for all $A \subseteq \Theta$.

C. Forecasting

Suppose now that we have some knowledge about the parameter $\theta \in \Theta$ after observing some training data $\mathbf{x}$. The goal is now to make a prediction from this knowledge as in [17]. The *forecasting* problem consists in making some predictions about some random quantity $Y \in \mathbb{Y}$ whose conditional distribution $g_{\mathbf{x},\theta}(y)$ given $\mathbf{X} = \mathbf{x}$ depends on θ. A belief function on $\mathbb{Y}$ can be derived from the sampling model proposed by Dempster [18]. For some unobserved auxiliary variable $Z \in \mathbb{Z}$ with known probability distribution μ independent of θ, we define a function $\varphi : \Theta \times \mathbb{Z} \to \mathbb{Y}$ so that

$$Y = \varphi(\theta, Z). \quad (12)$$

A multi-valued mapping $\Gamma'_{\mathbf{x}} : [0,1] \times \mathbb{Z} \to 2^{\mathbb{Y}}$ is defined by composing $\Gamma_{\mathbf{x}}$ with φ

$$\begin{array}{rccc} \Gamma'_{\mathbf{x}} : & [0,1] \times \mathbb{Z} & \to & 2^{\mathbb{Y}} \\ & (\gamma, z) & \mapsto & \varphi(\Gamma_{\mathbf{x}}(\gamma), z). \end{array} \quad (13)$$

A belief function on $\mathbb{Y}$ can then be derived from the product measure $\lambda \otimes \mu$ on $[0,1] \times \mathbb{Z}$ and the multi-valued mapping $\Gamma'_{\mathbf{x}}$

$$Bel_{\mathbf{x}}^{\mathbb{Y}}(A) = (\lambda \otimes \mu)\left(\{(\gamma, z) \mid \varphi\left(\Gamma_{\mathbf{x}}(\gamma), z\right) \subseteq A\}\right), \quad (14a)$$

$$Pl_{\mathbf{x}}^{\mathbb{Y}}(A) = (\lambda \otimes \mu)\left(\{(\gamma, z) \mid \varphi\left(\Gamma_{\mathbf{x}}(\gamma), z\right) \cap A \neq \emptyset\}\right), \quad (14b)$$

for all $A \subseteq \mathbb{Y}$.

III. Probabilistic multiclass calibration

Consider a multiclass classification problem with K classes and let $\mathbb{Y} = \{1, \ldots, K\}$ be the set of all possible class labels. Let $\mathcal{C}$ be a multiclass classifier trained to classify any instance of a feature space $\mathbb{X}$. Given a test data $\mathbf{x} \in \mathbb{X}$ with unknown label $y \in \mathbb{Y}$, the classifier returns a vector of scores $\mathbf{s} = \mathcal{C}(\mathbf{x}) \in \mathbb{R}^K$. The score $\mathbf{s}[k]$ represents the amount of support for the hypothesis $y = k$, for all $k \in \{1, \ldots, K\}$. In particular, the label predicted by $\mathcal{C}$ is given by $y^* = \arg\max_k \mathbf{s}[k]$.

The aim of probabilistic calibration is to transform the vector of scores $\mathbf{s} \in \mathbb{R}^K$ into a probability distribution $\mathbf{p} \in \mathcal{P}(\mathbb{Y})$ defined over $\mathbb{Y}$. The softmax function is commonly used for that purpose. In the general case of multinomial logistic regression, $\mathbf{p}$ is defined through a softmax function g defined as

$$\mathbf{p} = g(\mathbf{s}, \boldsymbol{\theta}) = [g_1(\mathbf{s}, \boldsymbol{\theta}), \ldots, g_K(\mathbf{s}, \boldsymbol{\theta})]^\top, \tag{15}$$

where

$$g_k(\mathbf{s}, \boldsymbol{\theta}) = \frac{\exp(\bar{\mathbf{s}}^\top \boldsymbol{\theta}_k)}{\sum_{j=1}^K \exp(\bar{\mathbf{s}}^\top \boldsymbol{\theta}_j)}, \quad \forall k \in \{1, \ldots, K\}. \tag{16}$$

The vector $\bar{\mathbf{s}} \in \mathbb{R}^{K+1}$ is the vector $\mathbf{s}$ concatenated with a constant value 1. The parameter $\boldsymbol{\theta} = [\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K] \in \mathbb{R}^{K+1 \times K}$ is unknown and needs to be estimated.

In a calibration context, several assumptions can be used in order to simplify the logistic regression problem. In equation (16), the quantity $\exp(\bar{\mathbf{s}}^\top \boldsymbol{\theta}_k)$ measures the support of the hypothesis $y = k$, the denominator being only a normalization factor. It would be reasonable to assume, as in [10], that this quantity only depends on the value of $\mathbf{s}[k]$ which also encodes the support of the same hypothesis. This implies that the parameter $\boldsymbol{\theta}$ can be written as

$$\boldsymbol{\theta} = \begin{bmatrix} a_1 & 0 & \cdots & 0 \\ 0 & a_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \ldots & 0 & a_K \\ b_1 & b_2 & \cdots & b_K \end{bmatrix}, \tag{17}$$

which yields

$$g_k(\mathbf{s}, \boldsymbol{\theta}) = \frac{\exp(a_k \mathbf{s}[k] + b_k)}{\sum_{j=1}^K \exp(a_j \mathbf{s}[j] + b_j)}, \ \forall k \in \{1, \ldots, K\}. \tag{18}$$

One can further impose that the calibration process should not change the ordering of the scores, i.e.,

$$\mathbf{s}[i] \geq \mathbf{s}[j] \Leftrightarrow g_i(\mathbf{s}, \boldsymbol{\theta}) \geq g_j(\mathbf{s}, \boldsymbol{\theta}), \ \forall i, j \in \{1, \ldots, K\}, \tag{19}$$

which leads to

$$a_1 = a_2 = \ldots = a_K \geq 0 \tag{20a}$$

$$\text{and} \quad b_1 = b_2 = \ldots = b_K. \tag{20b}$$

These equality constraints yield the following one-parameter functions:

$$g_k(\mathbf{s}, \theta) = \frac{\exp(\theta \mathbf{s}[k])}{\sum_{j=1}^K \exp(\theta \mathbf{s}[j])}, \ \forall k \in \{1, \ldots, K\}, \tag{21}$$

with θ taking value in $\Theta = [0, +\infty[$.

The parameter $\theta \in \Theta$ can be estimated given some training data. Let $\mathcal{X} = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n)\}$ be a set of n training data with $(\mathbf{x}_i, y_i) \in \mathbb{X} \times \mathbb{Y}$ and $\mathcal{S} = \{(\mathbf{s}_1, y_1), (\mathbf{s}_2, y_2), \ldots, (\mathbf{s}_n, y_n)\}$ be its associated set of scores where $\mathbf{s}_i = \mathcal{C}(\mathbf{x}_i)$. The optimal value of θ is determined by maximizing the likelihood function on the training set which is defined as follows:

$$L_{\mathcal{S}}(\theta) = \prod_{i=1}^n \left(\prod_{k=1}^K \mathbf{p}_i[y_k]^{t_{ik}} \right), \tag{22}$$

where

$$\mathbf{p}_i = g(\mathbf{s}_i, \boldsymbol{\theta}) \quad \text{and} \quad t_{ik} = \begin{cases} 1 & \text{if } y_i = k, \\ 0 & \text{otherwise.} \end{cases} \tag{23}$$

Maximizing the likelihood function (22) is equivalent to minimizing the negative log-likelihood function defined as

$$-\log(L_{\mathcal{S}}(\theta)) = -\sum_{i=1}^n \left(\sum_{k=1}^K t_{ik} \log(\mathbf{p}_i[y_k]) \right), \tag{24a}$$

$$= -\sum_{i=1}^n \sum_{k=1}^K t_{ik} \theta \mathbf{s}_i[k] \tag{24b}$$

$$+ \sum_{i=1}^n \left[\log \left(\sum_{j=1}^K \exp\left(\theta \mathbf{s}_i[j]\right) \right) \sum_{k=1}^K t_{ik} \right] \tag{24c}$$

which is a one-dimensional convex function. Minimizing (24) can thus be done using a simple iterative optimization algorithm. When the training data can be perfectly classified by $\mathcal{C}$, minimizing (24) will lead to an infinitely large value of θ. To avoid this situation, we propose to use an out-of-sample data model similar to the one introduced by Platt [1] for the binary case. Given that $y_i = k^*$, the coefficient t_{ik} is replaced by

$$t'_{ik} = \begin{cases} \frac{n_{k^*}+1}{n_{k^*}+K} & \text{if } k = k^*, \\ \frac{1}{n_{k^*}+K} & \text{otherwise,} \end{cases} \tag{25}$$

where n_{k^*} is the total number of training data of label k^*. Using this new formulation, the parameter $\widehat{\theta} \in \Theta$ maximizing the likelihood function (22) becomes unique and finite.

As reported by Xu et al. [4], the uncertainty of the estimated parameter $\widehat{\theta}$ is not taken into account in probabilistic calibration methods. In particular, the quality of the calibration is highly dependent of the number of training data. In Figure 1, we show the normalized likelihood function computed for the calibration of a multiclass SVM classifier trained on the Satimage dataset from the UCI repository [19]. When the number of training data increases, the estimated parameter becomes more accurate. By extending the probabilistic approach to the evidential framework, we aim at taking into account the whole shape of the likelihood function.

IV. Evidential multinomial calibration

The calibration method presented in the previous section can be extended to the evidential framework through three steps. First, the knowledge about the parameter $\theta \in \Theta$ is encoded by

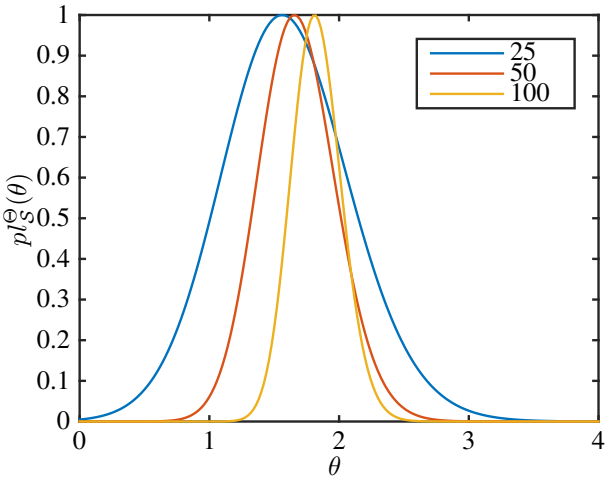


Fig. 1. Normalized likelihood function for different sample sizes.

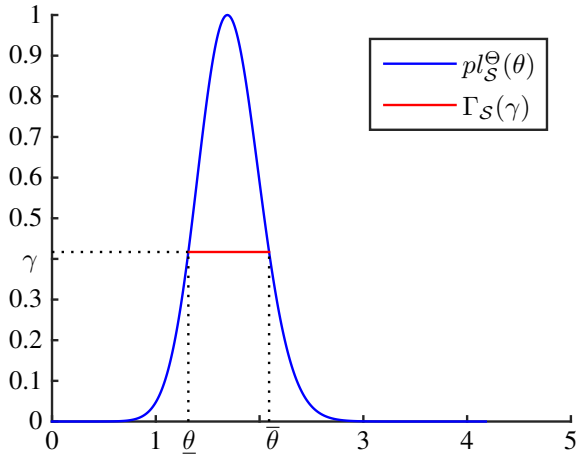


Fig. 2. For a level $\gamma \in [0,1]$, the associated level set is an interval $[\underline{\theta}, \overline{\theta}]$.

a belief function constructed from the likelihood function (22). Then, the unknown label associated to an observation is modeled by an auxiliary variable with known distribution. Finally, the information about θ is used to construct a predictive belief function.

A. Likelihood-based belief function

As explained in Section II-B, a belief function over the parameter $\theta \in \Theta$ can be constructed from the relative likelihood function. From the training data $\mathcal{S}$, we can build the following contour function:

$$pl^{\Theta}_{\mathcal{S}}(\theta) = \frac{L_{\mathcal{S}}(\theta)}{L_{\mathcal{S}}(\widehat{\theta})}, \quad \forall \theta \in \Theta. \tag{26}$$

A belief function $Bel^{\Theta}_{\mathcal{S}}$ and a plausibility function $Pl^{\Theta}_{\mathcal{S}}$ can be derived from this contour function as explained in Section II-B. In our particular case, the contour function $pl^{\Theta}_{\mathcal{S}}$ is unimodal, therefore the level sets of the belief function $Bel^{\Theta}_{\mathcal{S}}$ are closed intervals. For a given level $\gamma \in [0,1]$, the corresponding focal set $\Gamma_{\mathcal{S}}(\gamma)$ can be written as

$$\Gamma_{\mathcal{S}}(\gamma) = \{\theta \in \Theta \mid pl^{\Theta}_{\mathcal{S}}(\theta) \geq \gamma\} = [\underline{\theta}, \overline{\theta}]. \tag{27}$$

In Figure 2, the blue curve represents the contour function and the red segment corresponds to the focal set at the level γ. This example was computed when calibrating an SVM classifier trained using the Satimage dataset. In practice, there is often no closed form expression of $\Gamma_{\mathcal{S}}(\gamma)$. However, given the unimodal shape of the contour function, a simple numerical approach using dichotomy is efficient enough to approximate $\Gamma_{\mathcal{S}}(\gamma)$. Details of this numerical approximation are given in Algorithm 1.

B. Multinomial distribution model

In the case of a multiclass classification problem with K classes, the unknown label y of an observation can be seen as the realisation of a random variable Y with a K-categories multinomial distribution. Let $\mathbf{p} = (p_1, p_2, \ldots, p_K)$ be the parameter of this distribution, i.e., $\mathbb{P}(Y = k) = p_k$, for all $k \in \{1, \ldots, K\}$. Let Z be a random variable which has a uniform distribution in the interval $[0,1]$. The random variable Y can be generated from $\mathbf{p}$ and Z by the function φ defined as

$$\varphi(\mathbf{p}, Z) = k, \quad \text{with } P_{k-1} \leq Z < P_k, \tag{28}$$

where

$$P_k = \sum_{j=1}^{k} p_j \quad \text{and } P_0 = 0. \tag{29}$$

Algorithm 1 Level set approximation by dichotomy

Require: contour function $pl^{\Theta}_{\mathcal{S}}$, level γ, error tolerance ϵ
  `/* Lower bound computation */`
  $a \leftarrow 0$, $b \leftarrow \widehat{\theta}$, $c \leftarrow (a+b)/2$
  while $|pl^{\Theta}_{\mathcal{S}}(c) - \gamma| > \epsilon/2$ **do**
    if $pl^{\Theta}_{\mathcal{S}}(c) > \gamma$ **then** $b \leftarrow c$ **else** $a \leftarrow c$ **end if**
    $c \leftarrow (a+b)/2$
  end while
  $\underline{\theta} \leftarrow c$
  `/* Upper bound computation */`
  $a \leftarrow \widehat{\theta}$, $b \leftarrow 2\widehat{\theta}$, $c = (a+b)/2$
  while $pl^{\Theta}_{\mathcal{S}}(b) > \gamma$ **do** $b \leftarrow 2b$ **end while**
  while $|pl^{\Theta}_{\mathcal{S}}(c) - \gamma| > \epsilon/2$ **do**
    if $pl^{\Theta}_{\mathcal{S}}(c) > \gamma$ **then** $a \leftarrow c$ **else** $b \leftarrow c$ **end if**
    $c \leftarrow (a+b)/2$
  end while
  $\overline{\theta} \leftarrow c$
  return $\Gamma_{\mathcal{S}}(\gamma) \approx [\underline{\theta}, \overline{\theta}]$

Indeed, it can easily be verified that for all $k \in \{1, \ldots, K\}$ we have

$$\begin{aligned}\mathbb{P}(\varphi(\mathbf{p}, Z) = k) &= \mathbb{P}(P_{k-1} \leq Z < P_k) && \text{(30a)}\\ &= \mathbb{P}(Z < P_k) - \mathbb{P}(Z < P_{k-1}) && \text{(30b)}\\ &= P_k - P_{k-1} && \text{(30c)}\\ &= p_k && \text{(30d)}\end{aligned}$$

C. Predictive belief functions

Let $\mathbf{s}$ be the vector of scores associated with an observation of unknown label y. The probability distribution of the associated random variable Y is exactly determined by the knowledge of the parameter $\theta \in \Theta$. The parameter $\mathbf{p}$ of the multinomial distribution is given by $\mathbf{p} = g(\mathbf{s}, \theta)$. Given some knowledge about θ, a predictive belief function can be built as explained in Section II-C.

As we are using the optimistic decision rule, we do not need to compute the complete predictive belief function $Bel_{\mathbf{s}}^{\mathbb{Y}}$. We are only interested in the predictive contour function $pl_{\mathbf{s}}^{\mathbb{Y}}$ which is defined as follows:

$$pl_{\mathbf{s}}^{\mathbb{Y}}(j) = (\lambda \otimes \mu)(\{(\gamma, z) | j \in \varphi(g(\mathbf{s}, \Gamma_{\mathcal{S}}(\gamma)), z)\}), \quad (31)$$

for all $j \in \{1, \ldots, K\}$. The contour function $pl_{\mathbf{s}}^{\mathbb{Y}}$ can be approximated by Monte Carlo simulation. By drawing from a uniform distribution M independent pairs (γ_i, Z_i), $i = 1, \ldots, M$, the quantity $pl_{\mathbf{s}}^{\mathbb{Y}}(j)$ can be approximated by

$$\widehat{pl}_{\mathbf{s}}^{\mathbb{Y}}(j) = \frac{1}{M} \#\{i \in \{1, \ldots, M\} | j \in \varphi(g(\mathbf{s}, \Gamma_{\mathcal{S}}(\gamma_i)), Z_i)\}, \quad (32)$$

where the operator $\#$ corresponds to the cardinality. For each pair (γ_i, Z_i) and all $j \in \{1, \ldots, K\}$, it is necessary to test whether $j \in \varphi(g(\mathbf{x}, \Gamma_{\mathcal{S}}(\gamma_i)), Z_i)$. By writing $\Gamma_{\mathcal{S}}(\gamma_i) = [\underline{\theta}_i, \overline{\theta}_i]$, we have the following equivalence:

$$\begin{aligned}& j \in \varphi(g(\mathbf{x}, \Gamma_{\mathcal{S}}(\gamma_i)), Z_i) && \text{(33a)}\\ & \quad \Leftrightarrow \exists \theta \in [\underline{\theta}_i, \overline{\theta}_i],\ \varphi(g(\mathbf{s}, \theta), Z_i) = j && \text{(33b)}\\ & \quad \Leftrightarrow \exists \theta \in [\underline{\theta}_i, \overline{\theta}_i],\ G_{j-1}(\mathbf{s}, \theta) \leq Z_i < G_j(\mathbf{s}, \theta), && \text{(33c)}\end{aligned}$$

where

$$G_j(\mathbf{s}, \theta) = \sum_{k=1}^{j} g_k(\mathbf{s}, \theta) \quad \text{and} \quad G_0(\mathbf{s}, \theta) = 0. \quad (34)$$

Figure 3 illustrates the computation of (32). As in Figure 2, this example was computed from the calibration of an SVM classifier on the Satimage dataset which covers a set of six classes. Given a pair (γ, Z) and its associated interval $[\underline{\theta}, \overline{\theta}] = \Gamma_{\mathcal{S}}(\gamma)$, the set $\varphi(g(\mathbf{s}, \Gamma_{\mathcal{S}}(\gamma)), Z)$ is determined by the regions overlapped by Z. Within the gray area delimited by $[\underline{\theta}, \overline{\theta}]$, we can see that, for the given Z, only the three regions R_4, R_5 and R_6 are overlapped. Therefore, we have $\varphi(g(\mathbf{s}, \Gamma_{\mathcal{S}}(\gamma)), Z) = \{4, 5, 6\}$.

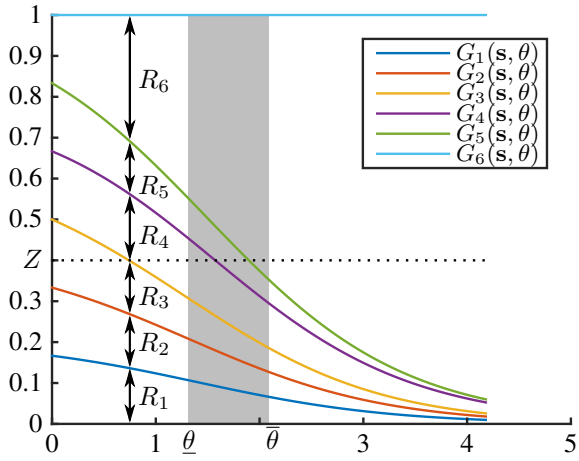


Fig. 3. For a given multinomial distribution of parameter $\mathbf{p} = g(\mathbf{s}, \theta)$, the value of $\varphi(\mathbf{p}, Z)$ is determined by the regions $R_1, \ldots, R_6$. The region R_j which is delimited by the curves $G_{j-1}(\mathbf{s}, \theta)$ and $G_j(\mathbf{s}, \theta)$ corresponds to the class j.

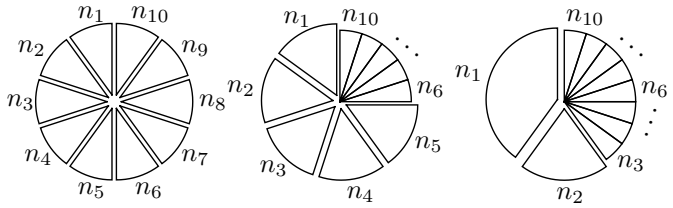
TABLE I
DATASETS FROM THE UCI REPOSITORY

Dataset	Number of classes	Training size	Testing size
Dna	3	1000	2186
Waveform	3	1000	4000
Satimage	6	1000	5435
Segment	7	1000	1310
Pendigits	10	1000	9992
USPS	10	1000	8292

V. EXPERIMENTAL EVALUATION

An experimental evaluation was conducted using six multiclass classification problems from the UCI repository [19]. For each of these datasets, the number of classes, the size of the training set and the size of the testing set are detailed in Table I. Similarly to [4], the calibration quality was evaluated by combining ten classifiers using three different scenarios as depicted in Figure 4. Let N be the total number of training data and $n_1, \ldots, n_{10}$ be the number of data used to train each of the ten classifiers, respectively. In particular, we have $n_1 + \ldots + n_{10} = N$. In the first scenario, the training set was uniformly partitioned into ten subsets, i.e., $n_1 = \ldots = n_{10} = N \times 10\%$. In the second one, the configuration was set to $n_1 = \ldots = n_5 = N \times 15\%$ and $n_6 = \ldots = n_{10} = N \times 5\%$. Finally, in the third scenario, the training set was partitioned using $n_1 = N \times 40\%$, $n_2 = N \times 20\%$ and $n_3 = \ldots = n_{10} = N \times 5\%$.

For each of the ten subsets, we trained and calibrated a multiclass SVM classifier. To get a vector of scores as output, we chose to train the classifier through a one-vs-all decomposition framework. Other multiclass SVM approaches such as the one proposed by Crammer and Singer [20] could also have been used. In [5], the authors used multiple cross-



(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

Fig. 4. Illustration of the three scenarios.

validation steps to set the parameters of the SVM and to calibrate it. In order to avoid potential bias and over-fitting, we adopted a more cautious approach. Each subset was further partitioned into two sets of equal size, the first part served as training data to learn the classifier while the second one was used as a validation set for parameter estimation and calibration.

To train the SVM classifiers, we used the LIBSVM library [21]. For the calibration, we used a quasi-Newton approach to minimize the negative log-likelihood (24). We used the implementation built within the function `fminunc` of MATLAB. Finally, for the evidential part, the level sets of $Bel_{\mathcal{S}}^{\Theta}$ were approximated with an error tolerance $\epsilon = 10^{-5}$ while the Monte Carlo simulation was conducted with $M = 10,000$.

The partitioning of the whole training set was itself generated randomly for 20 rounds. The average classification accuracy as well as the 95% confidence interval are reported in Table II. In our experiments, we compared the probabilistic approach proposed by Milgram et al. [10] to our evidential extension. We can see that for the Dna, Satimage, Segment and USPS datasets, our evidential method performed significantly better than the probabilistic one in all three scenarios. For the Waveform and Pendigits datasets, the probabilistic approach always performed better but was not significantly different from the evidential method except for the third scenario of the Pendigits dataset.

Except for the USPS dataset, the highest difference in accuracy was always reached in the third scenario. Similarly to results reported in [4], the evidential approach should be preferred when the classifiers to combine are trained with unbalanced amount of training data. However, in our case, even when all the ten classifiers were trained with the same number of data, the evidential calibration gave significantly better results for four of the datasets.

For the evidential approach, we can see that the best results were always obtained for the third scenario except for the Dna dataset. The evidential method accounted for the higher accuracy of the classifiers that were trained with a large number of data. At the same time, it efficiently encoded the higher uncertainty of those that were, in contrary, trained with fewer data. In contrast, for the probabilistic calibration, the results in the third scenario were always worse than in the second one. This can be explained by the fact the eight classifiers that were trained with very few data have the same amount of influence as the other two. The source code of the calibration method is available on the author's website[1].

VI. Conclusion

In this paper, we proposed an evidential multinomial logistic regression for multiclass classifier calibration. The use of belief functions can take into account the whole shape of the likelihood function. Therefore the uncertainty of the calibration step is better handled within the evidential framework.

The method proposed in this paper was applied to the calibration of SVM classifiers but it can also be applied to the calibration of other multiclass classifiers. Several probabilistic machine learning algorithms such as k-nearest-neighbors, decision trees or neural networks can directly return a multiclass probability distribution. Calibration of these types of classifiers and comparison to their evidential versions will be investigated in future work.

Acknowledgment

This research was carried out in the framework of the Labex MS2T, which was funded by the French Government, through the program "Investments for the future" managed by the National Agency for Research (Reference ANR-11-IDEX-0004-02). It was supported by the ANR-NSFC Sino-French PRETIV project (References ANR-11-IS03-0001 and NSFC-61161130528).

References

[1] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," in *Advances in Large-Margin Classifiers*, A. J. Smola, P. Bartlett, B. Schölkopf, and D. Schuurmans, Eds. MIT Press, 1999, pp. 61–74.

[2] B. Zadrozny and C. Elkan, "Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers," in *International Conference on Machine Learning*, Williamstown, Maryland, 2001, pp. 609–616.

[3] ——, "Transforming classifier scores into accurate multiclass probability estimates," in *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, USA: ACM, 2002, pp. 694–699.

[4] Ph. Xu, F. Davoine, and T. Denœux, "Evidential logistic regression for binary SVM classifier calibration," in *Belief Functions: Theory and Applications*, ser. Lecture Notes in Computer Science, F. Cuzzolin, Ed. Springer International Publishing, September 2014, vol. 8764, pp. 49–57.

[5] T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multiclass classification by pairwise coupling," *Journal of Machine Learning Research*, vol. 5, pp. 975–1005, 2004.

[6] T. Hastie and R. Tibshirani, "Classification by pairwise coupling," *The Annals of Statistics*, vol. 26, no. 1, pp. 451–471, 1998.

[7] B. Quost, T. Denœux, and M.-H. Masson, "Pairwise classifier combination in the framework of belief functions," in *Proceedings of the 8th International Conference on Information Fusion*, Philadelphia, USA, 2005.

[8] ——, "One-against-all classifier combination in the framework of belief functions," in *Proceedings of Information Processing and Management of Uncertainty in Knowledge-Based Systems*, Paris, France, July 2006, pp. 356–363.

[9] S. Destercke and B. Quost, "Correcting binary imprecise classifiers: Local vs global approach," in *Proceedings of the 6th International Conference on Scalable Uncertainty Management*, ser. Lecture Notes in Computer Science, E. Hüllermeier, S. Link, T. Fober, and B. Seeger, Eds., vol. 7520. Germany: Springer, September 2012, pp. 299–310.

[1] https://www.hds.utc.fr/~xuphilip/dokuwiki/en/data.

TABLE II

AVERAGE CLASSIFICATION ACCURACY WITH 95% CONFIDENCE INTERVAL. THE UNDERLINED NUMBERS CORRESPOND TO THE BEST RESULTS AND THE BOLD ONES ARE THOSE THAT ARE SIGNIFICANTLY BETTER. THE DIFFERENCE IN ACCURACY BETWEEN THE EVIDENTIAL AND PROBABILISTIC APPROACHES ARE REPORTED IN BRACKETS.

	Scenario 1		Scenario 2		Scenario 3	
Dataset	Probabilistic	Evidential	Probabilistic	Evidential	Probabilistic	Evidential
Dna	68.9 ± 1.7	$\underline{\mathbf{74.3 \pm 1.5}}$ $(+5.4)$	70.2 ± 1.6	$\underline{\mathbf{75.7 \pm 1.6}}$ $(+5.5)$	68.1 ± 1.5	$\underline{\mathbf{75.0 \pm 1.3}}$ $(+6.9)$
Waveform	$\underline{77.0 \pm 1.7}$	76.8 ± 1.7 (-0.2)	$\underline{76.5 \pm 1.9}$	76.1 ± 1.8 (-0.4)	$\underline{76.9 \pm 2.2}$	76.4 ± 2.1 (-0.5)
Satimage	78.5 ± 0.6	$\underline{\mathbf{80.3 \pm 0.9}}$ $(+1.8)$	79.1 ± 0.6	$\underline{\mathbf{81.2 \pm 0.8}}$ $(+2.1)$	78.6 ± 0.7	$\underline{\mathbf{81.2 \pm 0.8}}$ $(+2.6)$
Segment	90.5 ± 0.5	$\underline{\mathbf{91.4 \pm 0.4}}$ $(+0.9)$	90.9 ± 0.5	$\underline{\mathbf{91.8 \pm 0.4}}$ $(+0.9)$	90.0 ± 0.5	$\underline{\mathbf{91.8 \pm 0.4}}$ $(+1.8)$
Pendigits	$\underline{93.6 \pm 0.5}$	92.7 ± 0.5 (-0.9)	$\underline{94.2 \pm 0.4}$	93.3 ± 0.5 (-0.9)	$\underline{\mathbf{94.1 \pm 0.4}}$	93.1 ± 0.3 (-1.0)
USPS	84.9 ± 0.6	$\underline{\mathbf{86.5 \pm 0.6}}$ $(+1.6)$	86.1 ± 0.6	$\underline{\mathbf{87.3 \pm 0.6}}$ $(+1.2)$	86.0 ± 0.5	$\underline{\mathbf{87.9 \pm 0.4}}$ $(+1.9)$

[10] J. Milgram, M. Cheriet, and R. Sabourin, ""One against one" or "one against all": Which one is better for handwriting recognition with SVMs?" in *Tenth International Workshop on Frontiers in Handwriting Recognition*, G. Lorette, Ed., Université de Rennes 1. La Baule, France: Suvisoft, October 2006.

[11] G. Shafer, *A mathematical theory of evidence*. Princeton, New Jersey: Princeton University Press, 1976.

[12] H. T. Nguyen, *An introduction to random sets*. Boca Raton, Florida: Chapman and Hall/CRC press, 2006.

[13] P. Walley, *Statistical reasoning with imprecise probabilities*. Chapman and Hall, 1991, vol. 42.

[14] T. Denœux, "Analysis of evidence-theoretic decision rules for pattern classification," *Pattern Recognition*, vol. 30, no. 7, pp. 1095–1107, 1997.

[15] ——, "Likelihood-based belief function: justification and some extensions to low-quality data," *International Journal of Approximate Reasoning*, vol. 55, no. 7, pp. 1535–1547, 2014.

[16] ——, "Rejoinder on "Likelihood-based belief function: justification and some extensions to low-quality data"," *International Journal of Approximate Reasoning*, vol. 55, no. 7, pp. 1614–1617, 2014.

[17] O. Kanjanatarakul, S. Sriboonchitta, and T. Denœux, "Forecasting using belief functions: an application to marketing econometrics," *International Journal of Approximate Reasoning*, vol. 55, no. 5, pp. 1113–1128, 2014.

[18] A. Dempster, "The Dempster-Shafer calculus for statisticians," *International Journal of Approximate Reasoning*, vol. 48, no. 2, pp. 365–377, 2008.

[19] M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[20] K. Crammer and Y. Singer, "On the learnability and design of output codes for multiclass problems," *Machine Learning*, vol. 47, no. 2-3, pp. 201–233, 2002.

[21] C.-C. Chang and C.-J. Lin, "LIBSVM: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 27, pp. 1–27, 2011.