A study on majority-voting classifiers with guarantees on the probability of error ⋆

A. Carè * **M.C. Campi** * **F.A. Ramponi** * **S. Garatti** **
A.T.J.R. Cobbenhagen ***

* *Dept. of Information Engineering, University of Brescia, Brescia, Italy (e-mail: {algo.care,marco.campi,federico.ramponi}@unibs.it.*
** *Dept. of Electronics, Information and Bioengineering, Politecnico di Milano, Milano, Italy (e-mail: simone.garatti@polimi.it).*
*** *Dept. of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands (e-mail: a.t.j.r.cobbenhagen@tue.nl).*

Abstract: The Guaranteed Error Machine (GEM) is a classification algorithm that allows the user to set *a-priori* (i.e., before data are observed) an upper bound on the probability of error. Due to its strong statistical guarantees, GEM is of particular interest for safety critical applications in control engineering. Empirical studies have suggested that a pool of GEM classifiers can be combined in a majority voting scheme to boost the individual performances. In this paper, we investigate the possibility of keeping the probability of error under control in the absence of extra validation or test sets. In particular, we consider situations where the classifiers in the pool may have different guarantees on the probability of error, for which we propose a data-dependent weighted majority voting scheme. The preliminary results presented in this paper are very general and apply in principle to any weighted majority voting scheme involving individual classifiers that come with statistical guarantees, in the spirit of Probably Approximately Correct (PAC) learning.

Keywords: Classification, Machine learning, Multi-agent systems, Randomized methods, Optimisation.

1. INTRODUCTION

A binary classifier is a function $\widehat{y} : \mathbb{R}^k \to \{0,1\}$; in supervised classification the classifier is *trained* (i.e., it is suitably chosen by an algorithm) from a set of previously recorded pairs (x_i, y_i), where $x_i \in \mathbb{R}^k$ are vectors of *features* (e.g., health indicators extracted from a blood analysis; measures of a signal that enters a complex system; etc.) and $y_i \in \{0,1\}$ are *labels* denoting the corresponding class (e.g., *healthy/ill* in the case of a blood analysis; *bounded/unbounded* in the case of a system response). When fed with a new vector of features x the classifier $\widehat{y}(\cdot)$ provides an automated prediction $\widehat{y}(x)$ for the corresponding y. An important quantity to assess the quality of the classifier is the probability of error $\mathrm{PE}(\widehat{y})$, that is, the probability of the event $\{\widehat{y}(\mathbf{x}) \neq y\}$.

The Guaranteed Error Machine (GEM) is an algorithm to construct classifiers $\widehat{y}(\cdot)$ that was proposed in Campi [2010] and has a built-in mechanism to keep $\mathrm{PE}(\widehat{y})$ strictly under control, which makes it particularly attractive for safety critical applications (see e.g. Baronio et al. [2017]) and for control systems engineering when a quantitative assessment of sample-based control schemes is required (see e.g. Manganini et al. [2015]).

The bound on the error is achieved by adopting a ternary output $\{0, 1, \textit{unknown}\}$, so that the classifier can refrain from classification: in the case of an abstention, the classifier makes no error irrespective of the value of y. The key idea of GEM is that the complexity of a classifier can be tuned by the user by selecting the value of a parameter k. A large value of k leads to classifiers that more often return a 0 or 1 value, but these classifiers misclassify more frequently, whereas smaller values of k correspond to more risk-averse classifiers that return *unknown* with higher probability. An alternative use of GEM, that has been exploited and studied in Carè et al. [2018], is that of constructing a set of GEM classifiers for increasing values of k, and then selecting the one with the smallest value of k, say $\widehat{k}$, for which no *unknown*s are returned. The probability of error of the resulting classifier cannot be bounded *a-priori*, but the connection between $\widehat{k}$ and $\mathrm{PE}(\widehat{y})$ is so strong that the observed value of $\widehat{k}$ comes with an informative confidence interval for $\mathrm{PE}(\widehat{y})$. Such confidence interval is easily computed without resorting to any extra validation or test set, so that GEM classifiers are said to be "self-testing classifiers".

The construction of GEM comes with some degrees of freedom. For example, the construction starts from a data point (x_0, y_0) that is either known *a-priori* or randomly chosen from the available data set. A reasonable approach to reduce the arbitrariness of the choice is to consider all

⋆ Roy Cobbenhagen was supported by "Toeslag voor Topconsortia voor Kennis en Innovatie" (TKI HTSM) from the Ministry of Economic Affairs, the Netherlands.

A. Carè, M.C. Campi and F.A. Ramponi were supported by the H&W 2015 program of the University of Brescia under the project "Classificazione della fibrillazione ventricolare a supporto della decisione terapeutica" (CLAFITE).

the possible constructions simultaneously and then build a classifier based on majority voting among the available classifiers (which are called the *base classifiers*).

Although experimental studies in Manganini et al. [2015] suggest that majority voting classification schemes can actually boost the performance of GEM-like classifiers, it remains an open problem how to translate theoretical guarantees on the base classifiers to guarantees on the majority-voting classifier in such a way that they remain valid and practically meaningful: this is the problem that motivates our paper. Remarkably, GEM motivated our studies because of the tightness and practical usefulness of the resulting bounds, but, as we shall see, the results of this paper can be applied directly to any other type of base classifiers coming with confidence intervals on their probability of error in the spirit of Probably Approximately Correct (PAC) learning, see e.g. Graepel et al. [2005].

1.1 Contribution and structure of the paper

In Section 2 we introduce the mathematical framework of this paper and we observe that, at least in principle, majority voting schemes can worsen the performance with respect to the base classifiers. The conclusion is that one should aim at error estimation procedures that are able to detect whether or not the specific situation at hand is favourable to majority voting. We move some preliminary steps in this direction by connecting the error of the majority voting classifier to an agreement parameter. Section 3 is the main section of this paper, where we consider majority voting, and weighted majority voting, among base classifiers that are individually guaranteed to satisfy certain error thresholds at a given confidence level. We show the interesting fact that the proliferation of base classifiers does not necessarily lead to a loss of control on the probability of error. This is true even if the weights assigned to the base classifiers depend on the training set, under the condition that the voting scheme is adequately constrained. Conclusions are drawn in Section 4.

1.2 Existing literature and related research

The combination of classifiers into a voting-based classification scheme has a long tradition, and the literature on the topic is vast, in the wake of the celebrated gradient boosting machines and random forests, see Zhu [2015] for a review. On the theoretical side, we mention here the the notable studies of Schapire et al. [1998], Schapire and Freund [2012], where the generalization properties of majority voting schemes are related to the concept of margin in a statistical learning framework, and refer to Ruta and Gabrys [2002], Kuncheva et al. [2003], Kuncheva [2014] for studies on the best and worst cases. It is also worth observing that together with celebrated successes there came misconceptions: as it was noted in Vardeman and Morris [2013], the appeal to the Condorcet's Jury Theorem is a common but very weak and often misplaced argument to support voting decision schemes.

Our approach in this paper, similarly to Schapire et al. [1998] and differently from Ruta and Gabrys [2002], assumes that no validation set is available. On the other hand, differently from Schapire et al. [1998] and any other work that we are aware of, our focus is on transforming the existing guarantees for the base classifiers into guarantees for the voting classification scheme. At a conceptual level, the approach in our Section 3 has points in common with the so-called PAC-Bayes approaches, where a Kullback-Leibler divergence is used to penalise the departure from a reference distribution (which is called "prior" in the PAC-Bayes terminology). However, to apply the PAC-Bayes bounds of Lacasse et al. [2007] and Germain et al. [2015] the user needs either a validation set or the knowledge of the mechanism that generates the classifiers ("reconstruction function"). With our approach, instead, the user needs just to know the error thresholds for the base classifiers and their confidence level.

2. PRELIMINARY DEFINITIONS AND RESULTS

Let $\Delta = X \times Y$ denote the set of all possible data points (x, y), where x is the feature vector and y is a binary label, i.e. $y \in \{0, 1\}$ (a data point (x, y) will be denoted in compact form as δ). We assume that Δ is equipped with a probability measure $\mathbb{P}_\Delta$ *unknown to the user*.

A *training set* $\mathbb{T} \in \Delta^N$ is a random sample of data points $\delta^{(1)}, ..., \delta^{(N)}$, that we assume to be drawn according to the product measure $\mathbb{P}_\Delta^N$ (i.i.d. assumption).

A *classifier* $\widehat{y}$ is a map from X to $\{0, 1, \textit{unknown}\}$, and a *classification algorithm* $\mathcal{A}$ is a map from training sets $\mathbb{T}$ to classifiers. In the following we will consider an ensemble of classification algorithms $\{\mathcal{A}_c\}$, indexed by $c \in C$. Most often, C will be a finite set, $C = \{1, 2, ..., M\}$, but it can also be an infinite set. The classifier obtained by applying the classification scheme $\mathcal{A}_c$ to $\mathbb{T}$ will be denoted by $\widehat{y}_c$, and the dependence of $\widehat{y}_c$ from $\mathbb{T}$ will be left implicit.
We define the error function $\mathcal{E}(\delta, c)$ as follows:

$$\mathcal{E}(\delta, c) = \begin{cases} 0, & \text{if } \widehat{y}_c(x) = y \text{ or } \widehat{y}_c(x) = \textit{unknown}; \\ 1, & \text{if } \widehat{y}_c(x) \in \{0, 1\} \text{ and } \widehat{y}_c(x) \neq y. \end{cases}$$

The *probability of error* of a classifier $\widehat{y}_c$ is formally defined as

$$\mathrm{PE}(\widehat{y}_c) = \mathbb{E}_\Delta\left[\mathcal{E}(\delta, c)\right] = \int_\Delta \mathcal{E}(\delta, c)\ \mathrm{d}\mathbb{P}_\Delta(\delta).$$

In practice, it is seldom the case that the classifiers $\widehat{y}_c$, $c \in C$, agree unanimously on the label to assign to a feature vector x. Hence, to let a qualified majority emerge from the labels $\widehat{y}_c(x)$, $c \in C$, and produce an unambiguous answer in $\{0, 1\}$, for any given x let us introduce the partition $C = S_0(x) \cup S_1(x) \cup S_u(x)$, where $S_0(x) = \{c \in C : \widehat{y}_c(x) = 0\}$, $S_1(x) = \{c \in C : \widehat{y}_c(x) = 1\}$, and $S_u(x) = \{c \in C : \widehat{y}_c(x) = \textit{unknown}\}$. Furthermore, let us introduce a probability measure $\mathbb{P}_C$ on C. $\mathbb{P}_C$ is always the same irrespective of x and it is used to establish the majority in a more general sense than the usual, "democratic" one (the latter is recovered when $\mathbb{P}_C$ is uniform). We define the *voting majority* to be

$$S_*(x) = \begin{cases} S_0(x), & \text{if } \mathbb{P}_C(S_0(x)) \geq \mathbb{P}_C(S_1(x)), \\ S_1(x), & \text{if } \mathbb{P}_C(S_0(x)) < \mathbb{P}_C(S_1(x)), \end{cases} \tag{1}$$

and the *voting minority* as $S_-(x) = C - (S_*(x) \cup S_u(x))$. The majority classifier is now defined as follows.

Definition 1. (Majority classifier). For all x,

$$\widehat{y}_*(x) = \begin{cases} 0, & \text{if } \mathbb{P}_C(S_0(x)) \geq \mathbb{P}_C(S_1(x)), \\ 1, & \text{if } \mathbb{P}_C(S_0(x)) < \mathbb{P}_C(S_1(x)). \end{cases} \qquad \star$$

The definition yields a well-posed binary classifier irrespective of $\mathbb{P}_C(S_u(x))$, and takes into account, so to say, only the "actual voters". The fact that, in the case of a tie, we have assigned $\widehat{y}_*(x) = 0$ to be the voting majority's label (see equation (1)) is just for simplicity, and variations on the theme are clearly possible.

So far, the measure $\mathbb{P}_C$ should have been thought of as a weighing tool to compute the majority. Now $\mathbb{P}_C$ will be used to construct a peculiar, non-deterministic classifier that when asked to classify a point x first samples an index c from C according to $\mathbb{P}_C$ and then outputs the label $\widehat{y}_c(x)$. Such a classifier is known in the literature under the name of "Gibbs classifier".

Definition 2. (Gibbs classifier) We denote by $\widetilde{y}_C$ the random map $x \mapsto \widehat{y}_c(x)$, where c is a random variable distributed according to $\mathbb{P}_C$. ⋆

The role of the Gibbs classifier $\widehat{y}_C$ in this paper will be only instrumental to bounding $\mathrm{PE}(\widehat{y}_*)$. Observing that the Gibbs classifier misclassifies a given point δ with probability $\mathbb{E}_C[\mathcal{E}(\delta, c)]$, and given that δ and c are independent since c is always extracted from the same $\mathbb{P}_C$ for every x, its probability of error is naturally defined as $\mathrm{PE}(\widetilde{y}_C) = \mathbb{E}_\Delta[\mathbb{E}_C[\mathcal{E}(\delta, c)]]$. By an application of Fubini's theorem [1], we obtain the following useful equivalence:

$$\mathrm{PE}(\widetilde{y}_C) = \mathbb{E}_C[\mathbb{E}_\Delta[\mathcal{E}(\delta, c)]] = \mathbb{E}_C[\mathrm{PE}(\widehat{y}_c)]. \quad (2)$$

The following theorem relates $\mathrm{PE}(\widehat{y}_*)$ and $\mathrm{PE}(\widetilde{y}_C)$.

Theorem 1. Define $A = \inf_{x \in X} \mathbb{P}_C(S_*(x))$. It holds that

$$\mathrm{PE}(\widehat{y}_*) \leq \frac{1}{A} \cdot \mathrm{PE}(\widetilde{y}_C).$$

Proof. Letting $\mathbb{1}\{E\}$ be the indicator function of the event E, we have

$$\mathrm{PE}(\widetilde{y}_C) \geq \mathbb{E}_\Delta[\mathbb{E}_C[\mathcal{E}(\delta, c)\ \mathbb{1}\{c \in S_*(x)\}]]. \quad (3)$$

Note that, whenever $S_*(x)$ is non-empty, $\widehat{y}_*(x) = \widehat{y}_c(x)$ for all $c \in S_*(x)$; let c_x^* be any element of $S_*(x)$. Then, (3) is equal to

$$\begin{aligned}
&\mathbb{E}_\Delta[\mathbb{E}_C[\mathcal{E}(\delta, c_x^*)\ \mathbb{1}\{c \in S_*(x)\}]] \\
&\quad = \mathbb{E}_\Delta[\mathcal{E}(\delta, c_x^*)\ \mathbb{E}_C[\mathbb{1}\{c \in S_*(x)\}]] \\
&\quad \geq \mathbb{E}_\Delta\left[\mathcal{E}(\delta, c_x^*) \inf_{x \in X} \mathbb{E}_C[\mathbb{1}\{c \in S_*(x)\}]\right] \\
&\quad = \mathrm{PE}(\widehat{y}_*) \inf_{x \in X} \mathbb{P}_C(S_*(x)) = \mathrm{PE}(\widehat{y}_*)\ A.
\end{aligned}$$

□

The term A is an "agreement index" that quantifies the size of the voting majority in the worst case. The crucial fact for the theory here developed is that, since the distribution $\mathbb{P}_C$ is chosen by the user, such index is (at least in principle) an *observable quantity*. In the rest of the paper, the observable quantity A will allow us to express our results on the error of the majority classifier in a concise way. However, its global nature (A is computed as an infimum over the whole X) can be limiting in practice: the next two theorems, which can be proven similarly as Theorem 1, stand as alternative building blocks for a more practical approach where guarantees on the probability of error are based on the agreement of the base classifiers *at a given point* x.

[1] Throughout the paper we assume that the error function is measurable over $\Delta \times C$ with respect to the product measure $\mathbb{P}_\Delta \times \mathbb{P}_C$.

Theorem 2. Define $A_x = \mathbb{P}_C(S_*(x))$ and let $a \in (0, 1]$. It holds that

$$\mathbb{P}_\Delta\{\widehat{y}_*(x) \neq y \wedge A_x \geq a\} \leq \frac{1}{a} \cdot \mathrm{PE}(\widetilde{y}_C).$$

⋆

Theorem 3. Consider the case where the base classifiers deliver no *unknown*s. Assume that $\mathbb{P}_\Delta\{A_x \in [\underline{a}, \bar{a}]\} \geq \underline{\nu}$, with $0.5 < \underline{a} \leq \bar{a} \leq 1$. Then,

$$\mathbb{P}_\Delta\{\widehat{y}_*(x) \neq y \wedge A_x \in [\underline{a}, \bar{a}]\} \leq \frac{\mathrm{PE}(\widetilde{y}_C) - (1 - \bar{a})\underline{\nu}}{2\underline{a} - 1}.$$

⋆

Going back to Theorem 1, we remark that, without restrictions on the probability with which $\widehat{y}_c(x) = \textit{unknown}$, the misclassification probability can worsen to an arbitrary extent with respect to the performance of the individual classifiers. This is not surprising as we are forcing the majority classifier to be a binary classifier, so that for values of x such that almost all the classifiers are uncertain, a decision is made based on the voice of a small subset of them that could always be wrong. We remark also that, if there are no *unknown*s in the set $\{\widehat{y}_c(x)\colon c \in C, x \in X\}$, then we necessarily have $A \geq \frac{1}{2}$, from which the well-known bound $\mathrm{PE}(\widehat{y}_*) \leq 2\mathrm{PE}(\widetilde{y}_C)$ is obtained. It can be shown that this bound is tight and there are situations where the performance of the majority worsens with respect to the performance of any individual classifier. Such possibility calls for mathematical results that enable the user to detect whether or not the situation at hand is favourable to majority voting.

3. A POOL OF GUARANTEED CLASSIFIERS

We start by considering the situation where the probability of error of each classifier $\widehat{y}_c$ is bounded by a given ϵ_c. From equation (2) and Theorem 1 it follows immediately that

$$\mathrm{PE}(\widehat{y}_*) \leq \frac{1}{A}\mathbb{E}_C[\epsilon_c]. \quad (4)$$

This simple relation suggests that, when some classifiers come with small ϵ_c while others come with a large ϵ_c, it might be advisable to concentrate the distribution $\mathbb{P}_C$, according to which the majority is computed, on the best guaranteed classifiers. [2] For example, we might want to define $\mathbb{P}_C$ so that it concentrates on the classifier with minimum ϵ_c, say $\bar{\epsilon}$; then, assuming there is a unique such classifier and there are no *unknown*s, the threshold given by the right-hand side of (4) becomes equal to $\bar{\epsilon}$, being $A = 1$ by construction.

The possibility of choosing $\mathbb{P}_C$ according to the thresholds ϵ_c is complicated by the fact that, in real-life machine learning, the values of ϵ_c are typically stochastic estimates that are not valid with absolute certainty but only with some confidence. In particular, we study here the situation where the ϵ_c are functions of the training set $\mathbb{T}$, as it is expressed by the following condition.

Condition 1. For all $c \in C$, there exists a function $\epsilon_c : \Delta^N \to [0, 1]$ such that

$$\mathbb{P}_\Delta^N\{\mathrm{PE}(\widehat{y}_c) > \epsilon_c(\mathbb{T})\} \leq \beta. \quad (5)$$

[2] This reminds us of a famous quote from Cicero's: *Et vero in dissensione civili, cum boni plus quam multi valent, expendendos cives, non numerandos puto.* (De re publica VI.1, ed. J.G.F. Powell.)

Remark 1. Clearly, Condition 1 can be trivially met by any classification scheme that divides the training set into two parts and uses only the first part to train the classifier, while the second part is used to estimate the probability of error at a sufficient confidence level. More interestingly, Condition 1 is met, with values of ϵ_c and β that are often satisfactory, by a modified version of the GEM algorithm of Campi [2010], where the *a-priori* selected complexity parameter k is increased by 1 until there are no more *unknowns*[3], see Carè et al. [2018] for an application of this idea. Moreover, it is satisfied by classification schemes to which the more general theory of the Wait-and-Judge scenario approach (Campi and Garatti [2018], Carè et al. [2019]) can be applied to relate probability of error and complexity of the classifier.[4] Also compression schemes, see Graepel et al. [2005], Margellos et al. [2015], Campi et al. [2018], satisfy such a condition, although the resulting bounds are often more conservative and less applicable than with GEM. In all of these cases, the dependence of the error threshold ϵ_c on $\mathbb{T}$ is typically reduced to the dependence on a complexity parameter $\widehat{k}$ that can be easily computed from $\mathbb{T}$. $\star$

For short, we will denote by $\widehat{\epsilon}_c$ the value $\epsilon_c(\mathbb{T})$. We now assume that $\mathbb{P}_C$ is a $\mathbb{T}$-dependent probability measure that the user can choose at will, for example, but not necessarily, by using the information carried by the values of $\widehat{\epsilon}_c$.

When C is finite, the following proposition can be often used to ensure that (4) holds true with high confidence

Proposition 1. Under Condition 1, if C is finite, $C = \{1, ..., M\}$, then

$$\mathbb{P}_\Delta^N \left\{ \mathrm{PE}(\widehat{y}_*) \leq \frac{1}{A}\mathbb{E}_C\left[\widehat{\epsilon}_c\right] \right\} \geq 1 - M\beta. \qquad (6)$$

Proof. Clearly, $\mathbb{P}_\Delta^N\{\mathrm{PE}(\widehat{y}_c) \leq \mathbb{E}_C\left[\widehat{\epsilon}_c\right]\} \geq \mathbb{P}_\Delta^N\{\mathrm{PE}(\widehat{y}_c) \leq \widehat{\epsilon}_c \ \forall c \in C\} = 1 - \mathbb{P}_\Delta^N\{\exists c \in C \text{ s.t. } \mathrm{PE}(\widehat{y}_c) > \widehat{\epsilon}_c\}$. By a simple union bound argument, $\mathbb{P}_\Delta^N\{\exists c \in C \text{ s.t. } \mathrm{PE}(\widehat{y}_c) > \widehat{\epsilon}_c\} \leq \sum_{c=1}^M \mathbb{P}_\Delta^N\{\mathrm{PE}(\widehat{y}_c) > \widehat{\epsilon}_c\}$ and the claim follows by Condition 1. $\square$

However, for a large number M of classifiers, the confidence value $1 - M\beta$ ensured by (6) becomes easily too low, and indeed uninformative.

The next result shows that, by enforcing a constraint on the way in which $\mathbb{P}_C$ can vary as a function of $\mathbb{T}$, it is still possible to keep under control the probability of error of the majority even when C is too large for (6) to be useful. In particular, the following theorem shows that if $\mathbb{P}_C$ is prevented from concentrating too much (as compared to a reference distribution $\bar{\mathbb{P}}_C$), then meaningful confidence levels can be achieved also when C is infinite.

Theorem 4. Assume that there exists a probability measure $\bar{\mathbb{P}}_C$ that does not depend on $\mathbb{T}$ and a $\bar{\kappa} > 0$ such that $\mathbb{P}_C(E) \leq \bar{\kappa}\bar{\mathbb{P}}_C(E)$ for all the measurable sets $E \subseteq X$ and every $\mathbb{T}$. Then, under Condition 1, for $\varrho > 0$ it holds that

$$\mathbb{P}_\Delta^N \left\{ \mathrm{PE}(\widehat{y}_*) \leq \frac{1}{A}(\mathbb{E}_C\left[\widehat{\epsilon}_c\right] + \varrho) \right\} \geq 1 - \frac{\bar{\kappa}}{\varrho} \cdot \beta. \qquad (7)$$

Proof. By Markov's inequality it holds that $\mathbb{P}_\Delta^N\{\mathrm{PE}(\widetilde{y}_C) - \mathbb{E}_C\left[\widehat{\epsilon}_c\right] > \varrho\} \leq \frac{1}{\varrho}\mathbb{E}_{\Delta^N}\left[(\mathrm{PE}(\widetilde{y}_C) - \mathbb{E}_C\left[\widehat{\epsilon}_c\right])^+\right]$, where $(x)^+ = \max\{x, 0\}$. By (2) and by Jensen's inequality ($(\cdot)^+$ is convex), we get $\frac{1}{\varrho}\mathbb{E}_{\Delta^N}\left[(\mathrm{PE}(\widetilde{y}_C) - \mathbb{E}_C\left[\widehat{\epsilon}_c\right])^+\right] \leq \frac{1}{\varrho}\mathbb{E}_{\Delta^N}\left[\mathbb{E}_C\left[(\mathrm{PE}(\widehat{y}_c) - \widehat{\epsilon}_c)^+\right]\right]$. Bounding the $\mathbb{T}$-dependent $\mathbb{P}_C$ with the $\mathbb{T}$-independent $\bar{\kappa}\bar{\mathbb{P}}_C$ and denoting $\bar{\mathbb{E}}_C[\cdot]$ the expectation with respect to $\bar{\mathbb{P}}_C$, we get

$$\begin{aligned}
&\frac{1}{\varrho}\mathbb{E}_{\Delta^N}\left[\mathbb{E}_C\left[(\mathrm{PE}(\widehat{y}_c) - \widehat{\epsilon}_c)^+\right]\right] \\
&\leq \frac{1}{\varrho}\mathbb{E}_{\Delta^N}\left[\bar{\kappa}\bar{\mathbb{E}}_C\left[(\mathrm{PE}(\widehat{y}_c) - \widehat{\epsilon}_c)^+\right)\right] \\
&= \frac{\bar{\kappa}}{\varrho}\bar{\mathbb{E}}_C\left[\mathbb{E}_{\Delta^N}\left[(\mathrm{PE}(\widehat{y}_c) - \widehat{\epsilon}_c)^+\right)\right] \\
&\quad [\text{by using } \mathrm{PE}(\widehat{y}_c) - \widehat{\epsilon}_c \leq 1] \\
&\leq \frac{\bar{\kappa}}{\varrho}\bar{\mathbb{E}}_C\left[\mathbb{P}_\Delta^N\{\mathrm{PE}(\widehat{y}_c) > \widehat{\epsilon}_c\}\right] \leq \frac{\bar{\kappa}}{\varrho}\beta,
\end{aligned}$$

where the last inequality is by Condition 1. Then, Theorem 1 can be invoked to claim that $\mathrm{PE}(\widehat{y}_*) \leq \frac{1}{A}(\mathbb{E}_C\left[\widehat{\epsilon}_c\right] + \varrho)$ for all $\mathbb{T} \in \Delta^N$ except for a subset with probability at most $\frac{\bar{\kappa}}{\varrho}\beta$, which was the claim to be proven. $\square$

When $\bar{\mathbb{P}}_C$ is uniform, $\bar{\kappa}$ can be interpreted as a parameter that balances between strict democracy and oligarchy: when $C = \{1, ..., M\}$, $\bar{\kappa} = 1$ enforces $\mathbb{P}_C$ to be uniform (democracy): the confidence value is maximum $(1 - \frac{\beta}{\varrho})$ but $\mathbb{E}_C\left[\widehat{\epsilon}_c\right]$ averages over both high and low values of $\widehat{\epsilon}_c$; on the other hand, $\bar{\kappa} = M$ does not constrain $\mathbb{P}_C$ at all, so that one is free to choose the measure $\mathbb{P}_C$ that concentrates all the probability on a single classifier with the smallest $\widehat{\epsilon}_c$: in this case, $\mathbb{E}_C\left[\widehat{\epsilon}_c\right]$ is minimized but the confidence becomes $1 - \frac{M}{\varrho}\beta$. Choosing $\bar{\kappa} = j$, $j \in \{2, ..., M-1\}$ leads to intermediate situations where the probability is distributed over a subset of j classifiers (e.g., the j classifiers with smallest $\widehat{\epsilon}_c$). A similar reasoning applies also to the case where $\mathbb{P}_C$ has density.

Example 1. (Numerical Example). Suppose that $M = 10000$ classifiers are given, each coming with a guarantee like the one in Condition 1 where $\beta = 10^{-4}$, and let $\varrho = 0.02$. If $\mathbb{P}_C$ can concentrate on a single classifier, our results do not provide any meaningful confidence as $M\beta = 1$. However, by setting $\bar{\mathbb{P}}_C(c) = \frac{1}{10000}$ for all c (uniform distribution) and $\bar{\kappa} = 2$, we are allowed to build a majority classifier based on the 5000 $(= M/\bar{\kappa})$ classifiers with the smallest $\widehat{\epsilon}_c$ and discharge the vote of the remaining 5000. Indeed, a probability measure $\mathbb{P}_C$ having uniform mass over the 5000 classifiers with the smallest $\widehat{\epsilon}_c$ and 0

[3] We emphasise the case where there are no *unknowns* because, in this case, the fact that $A \geq 1/2$ impedes the deterioration of our bounds. However, the results of this section are valid also in the presence of *unknowns*. Interestingly, by allowing *unknowns* it is possible to satisfy Condition 1 with a function $\epsilon_c(\mathbb{T})$ that is constant with respect to $\mathbb{T}$, by resorting to GEM in its original form, Campi [2010], or by stopping the construction of the algorithm Carè et al. [2018] after a predetermined number of steps. Clearly, with *unknowns*, the values of ϵ_c provide no immediate indication about how to unbalance $\mathbb{P}_C$.

[4] We notice the subtle fact that, with GEM, iteratively increasing k by one until a $\widehat{k}$ is reached such that there are no *unknowns* is not equivalent to setting $k = \infty$ and then counting the number of actual support points. While in the first case guarantees for the final classifier can be obtained by a simple union bound for all the possible values of $\widehat{k}$, in the latter case one should resort to the Wait-and-Judge theory, see Appendix 1 in Campi and Garatti [2018].

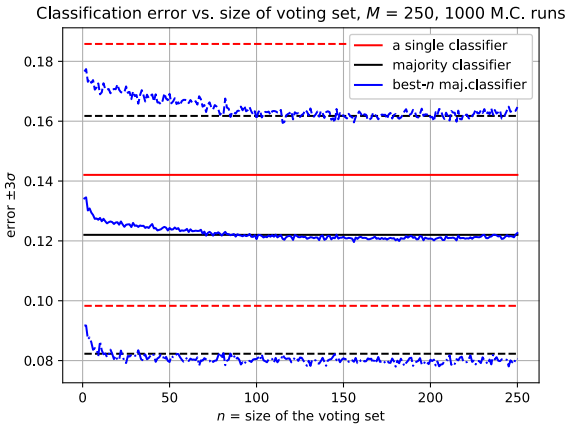

Fig. 1. Empirical classification error $\widehat{\mathrm{pe}}(n)$, $n = 1, \ldots, 250$. (Solid lines: average over 1000 Monte Carlo runs; dashed lines: average $\pm$ 3 standard deviations estimated from the same sample.) In red: avg. $\pm\, 3\sigma$ for a single classifier. In black: avg. $\pm\, 3\sigma$ of the majority classifier taking into account *all* the classifiers ($\mathbb{P}_C$ = uniform). In blue: avg. $\pm\, 3\sigma$ of the majority classifier taking into account *the "best"* n classifiers ($\mathbb{P}_C$ = uniform over the n classifiers with smallest $\widehat{\epsilon}_c$).

mass over all the other classifiers satisfies $\mathbb{P}_C\{c\} \leq 2 \cdot \bar{\mathbb{P}}_C(c)$ for all c; hence Theorem 4 ensures a confidence of 99%.

Example 2. (Simulations). We ran a preliminary campaign of simulations with the following setup: a pool of $M = 250$ classifiers built according to the scheme of Carè et al. [2018] was trained over the same dataset of 250 independent data points (x_i, y_i), where each $x_i = (x_{1i}, x_{2i})$, $i = 1, \ldots, 250$, was chosen according to the uniform distribution over $[0, 1]^2$, and y_i was distributed according to the following function:

$$y_i = y(x_i) = \begin{cases} 1, & \text{if } x_{2i} \geq \left(x_{1i} - \frac{1}{2}\right) \cos\left(25 x_{1i}\right) + \frac{1}{2} \\ 0, & \text{otherwise} \end{cases}$$

(the same function as in the experimental section of Cobbenhagen et al. [2019]); each classifier was trained having the i-th point of the dataset as the starting point and the $N = M - 1 = 249$ remaining ones as the training set. Since these classifiers come with different values of $\widehat{\epsilon}_c$ that are individually valid with confidence $1 - \beta$, the results of this section apply: we sorted the classifiers in order of increasing $\widehat{\epsilon}_c$, we chose a $\mathbb{P}_C$ uniform over the "voting set" of the "best" n classifiers (i.e. those with the smallest $\widehat{\epsilon}_c$), and we tested the resulting majority classifier $\widehat{y}_*$ on $T = 10000$ random points extracted according to the same distribution of the training set, recording the average error $\widehat{\mathrm{pe}}(n) = \#\mathrm{errors}/T$. We repeated the entire procedure for all the possible sizes $n = 1, 2, \ldots, 250$ of the voting set, and 1000 times for each size n in a Monte Carlo simulation. The results are summarized in Fig. 1. As the reader can see from this simulation, and as we have ascertained from other ones, restricting the classifiers to a very small voting set is not convenient; on the other hand, there seems to be a tendency to improvement as the size increases, that settles around *half* the number of classifiers. This fact is coherent with the trade-off between error and confidence expressed by Theorem 4: Fig. 2 shows the values

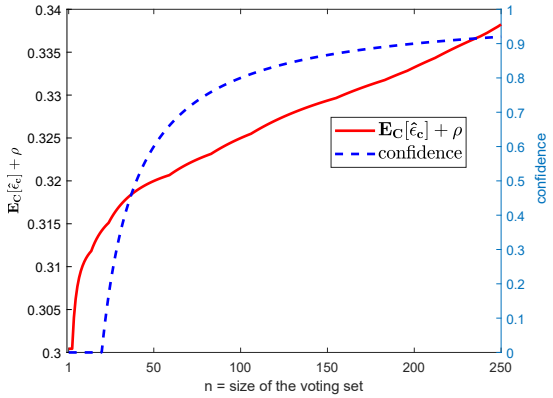

Fig. 2. Error-confidence trade-off as given by an application of Theorem 4 to typical values of $\{\widehat{\epsilon}_c\}$, for different n, when $\beta = 4 \cdot 10^{-3}$, $\varrho = 0.05$.

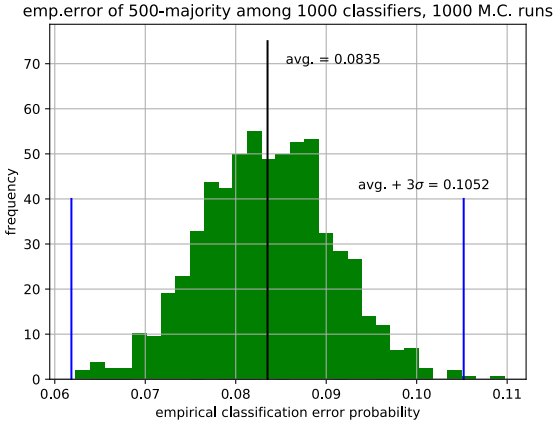

Fig. 3. Histogram of empirical classification errors for a 500-majority classifier ($\mathbb{P}_C$ = uniform over the classifiers with the $n = 500$ smallest $\widehat{\epsilon}_c$) in a pool of $M = 1000$ classifiers trained with 1000 samples.

of $\mathbb{E}[\widehat{\epsilon}_c] + \varrho$ (left ordinate) and of the confidence $1 - \frac{\kappa}{\varrho}\beta$ (right ordinate) when Theorem 4 is applied to a random outcome of the experiments at hand: the "best" classifiers in terms of $\mathbb{E}_C[\widehat{\epsilon}_c]$ are weaker in terms of confidence, i.e., their promising threshold can be exceeded with higher probability.

Repeated simulations in the same setup as above but with a larger dataset of size 1000 and $M = 1000$ classifiers were also performed. To apply Theorem 4 we fixed $\beta = 10^{-4}$ and $\varrho = 0.01$. The experimental results for the voting size $n = 500$ are shown in Figs. 3 and 4. The voting size $n = 500$ corresponds to choosing $\bar{\kappa} = 2$; from the results of the Monte Carlo simulation we obtain that $\mathbb{E}_C[\widehat{\epsilon}_c]$ is, on average, 14.22%. Then, the "typical" upper-bound that is issued by using the formula $\frac{1}{A}(\mathbb{E}_C[\widehat{\epsilon}_c] + \varrho)$ in (7) is $\mathrm{PE}(\widehat{y}_*) \leq 2(14.22\% + 1\%) = 30.44\%$ (where, for simplicity, we have substituted A with its lower bound $A \geq \frac{1}{2}$), to which a confidence of $1 - \frac{2\beta}{\varrho} = 98\%$ is attached. By comparing Figs. 3 and 4, one can note that, in 1000 runs, the upper-bounds $\frac{1}{A}(\mathbb{E}_C[\widehat{\epsilon}_c] + \varrho)$ are always higher than the empirical errors. On the one hand this confirms the soundness of the approach; on the other hand it suggests that the bounds hold with a probability higher than the

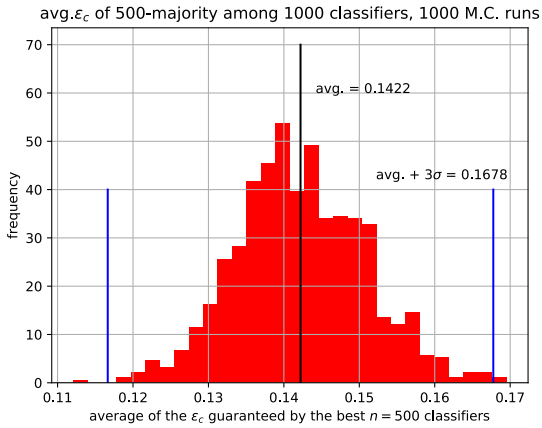

Fig. 4. Histogram of the values of $\mathbb{E}_C[\widehat{\epsilon}_c]$ for the 500-majority classifier of Fig. 3. The $\widehat{\epsilon}_c$ of each classifier c in the 500-set is computed according to the theory in Carè et al. [2018] corresponding to $\beta = 10^{-4}$.

98% probability guaranteed by Theorem 4: this reveals some conservatism that calls for further investigation. In particular, an important source of conservatism is expected to be removed by exploiting local values A_x in the vein of Theorems 2 and 3 (in this respect, it is remarkable that $\mathbb{E}_\Delta[A(x)]$ was always above 90% in our Monte Carlo runs).

4. CONCLUSIONS AND FUTURE STUDIES

In this paper, we have investigated the problem of transforming individual PAC guarantees for base classifiers into guarantees for majority voting decision schemes. We have discussed weighted majority decision schemes where the weights can depend on the training set, and we have shown that, by suitably restricting the variability of the weights, guarantees can be provided even when the base classifiers are infinitely many. While the bounds in this paper are expressed in terms of a worst-case agreement index, it is possible to provide bounds that are differentiated based on the agreement level for the observed feature vector, in the line of Theorems 2 and 3. In future works, we will aim at exploiting the specific construction of the base classifiers to improve the guarantees; for example, estimators based on the number of support points, like those in Cobbenhagen et al. [2019], could be used to detect situations that are more favourable to majority voting.

While in this paper we focused on bounding the probability of error computed with respect to all the possible pairs (x, y), we plan to study the probability of error *conditional to* the level of agreement among the base classifiers. Preliminary studies in this direction can be found in Cobbenhagen et al. [2019].

The extension of the framework of this paper to the regression problem, in the line of Garatti and Carè [2019], is also the subject of current research.

REFERENCES

Baronio, F., Baronio, M., Campi, M.C., Carè, A., Garatti, S., and Perone, G. (2017). Ventricular defibrillation: Classification with G.E.M. and a roadmap for future investigations. In *2017 IEEE 56th CDC*, 2718–2723. doi:10.1109/CDC.2017.8264054.

Campi, M.C., Garatti, S., and Ramponi, F.A. (2018). A general scenario theory for nonconvex optimization and decision making. *IEEE Trans. Automat. Contr.*, 63(12), 4067–4078. doi:10.1109/TAC.2018.2808446.

Campi, M.C. (2010). Classification with guaranteed probability of error. *Mach. Learn.*, 80, 63–84. doi:10.1007/s10994-010-5183-x.

Campi, M.C. and Garatti, S. (2018). Wait-and-judge scenario optimization. *Math. Program.*, 167(1), 155–189. doi:10.1007/s10107-016-1056-9.

Carè, A., Garatti, S., and Campi, M.C. (2019). The wait-and-judge scenario approach applied to antenna array design. *Comput. Manag. Sci.* doi:10.1007/s10287-019-00345-5.

Carè, A., Ramponi, F.A., and Campi, M.C. (2018). A new classification algorithm with guaranteed sensitivity and specificity for medical applications. *IEEE Control Syst. Lett.*, 2(3), 393–398. doi:10.1109/LCSYS.2018.2840427.

Cobbenhagen, R., Carè, A., Campi, M., Ramponi, F., and Heemels, M. (2019). Consensus and reliability: The case of two binary classifiers. In *Proc. of the 8th IFAC NECSYS 2019, Chicago, IL, USA*.

Garatti, S. Campi, M. and Carè, A. (2019). On a class of interval predictor models with universal reliability. *Automatica*, 110, 108542. doi:10.1016/j.automatica.2019.108542.

Germain, P., Lacasse, A., Laviolette, F., Marchand, M., and Roy, J.F. (2015). Risk bounds for the majority vote: From a PAC-Bayesian analysis to a learning algorithm. *J. Mach. Learn. Res.*, 16, 787–860.

Graepel, T., Herbrich, R., and Shawe-Taylor, J. (2005). PAC-Bayesian Compression Bounds on the Prediction Error of Learning Algorithms for Classification. *Mach. Learn.*, 59(1-2), 55–76. doi:10.1007/s10994-005-0462-7.

Kuncheva, L., Whitaker, C., Shipp, C., and Duin, R. (2003). Limits on the majority vote accuracy in classifier fusion. *Pattern Analysis & Applications*, 6(1), 22–31. doi:10.1007/s10044-002-0173-7.

Kuncheva, L.I. (2014). *Combining Pattern Classifiers: Methods and Algorithms, 2nd Edition*. Wiley.

Lacasse, A., Laviolette, F., Marchand, M., Germain, P., and Usunier, N. (2007). PAC-Bayes Bounds for the Risk of the Majority Vote and the Variance of the Gibbs Classifier. In *Adv. Neural Inf. Process. Syst. 19*, 1, 769–776. The MIT Press. doi:10.7551/mitpress/7503.003.0101.

Manganini, G., Piroddi, L., and Prandini, M. (2015). A classification-based approach to the optimal control of affine switched systems. In *2015 54th IEEE CDC*, 2963–2968. doi:10.1109/CDC.2015.7402667.

Manganini, G., Falsone, A., and Prandini, M. (2015). A majority voting classifier with probabilistic guarantees. In *2015 IEEE CCA*, 1084–1089. IEEE, Sydney, Australia. doi:10.1109/CCA.2015.7320757.

Margellos, K., Prandini, M., and Lygeros, J. (2015). On the connection between compression learning and scenario based single-stage and cascading optimization problems. *IEEE Trans. Automat. Contr.*, 60(10), 2716–2721. doi:10.1109/TAC.2015.2394874.

Ruta, D. and Gabrys, B. (2002). A theoretical analysis of the limits of majority voting errors for multiple classifier systems. *Pattern Analysis and Applications*, 5(4), 333–350. doi:10.1007/s100440200030.

Schapire, R. and Freund, Y. (2012). *Boosting: Foundations and Algorithms*. Adaptive computation and machine learning. MIT Press.

Schapire, R.E., Freund, Y., Bartlett, P., Lee, W.S., et al. (1998). Boosting the margin: A new explanation for the effectiveness of voting methods. *The annals of statistics*, 26(5), 1651–1686.

Vardeman, S.B. and Morris, M.D. (2013). Majority voting by independent classifiers can increase error rates. *The American Statistician*, 67(2), 94–96. doi:10.1080/00031305.2013.778788.

Zhu, M. (2015). Use of majority votes in statistical learning. *Wiley Interdisciplinary Reviews: Computational Statistics*, 7(6), 357–371. doi:10.1002/wics.1362.