Diversity-Aware Weighted Majority Vote Classifier for Imbalanced Data

Anil Goyal
NEC Laboratories Europe GmbH
Heidelberg, Germany
anil.goyal@neclab.eu

Jihed Khiari
NEC Laboratories Europe GmbH
Heidelberg, Germany
jihed.khiari@neclab.eu

Abstract—In this paper, we propose a diversity-aware ensemble learning based algorithm, referred to as `DAMVI`, to deal with imbalanced binary classification tasks. Specifically, after learning base classifiers, the algorithm i) increases the weights of positive examples (minority class) which are "hard" to classify with uniformly weighted base classifiers; and ii) then learns weights over base classifiers by optimizing the PAC-Bayesian C-Bound that takes into account the accuracy and diversity between the classifiers. We show efficiency of the proposed approach with respect to state-of-art models on predictive maintenance tasks, credit card fraud detection, webpage classification and medical applications.

Index Terms—Imbalanced Data, Ensemble Learning, C-Bound, Diversity

I. INTRODUCTION

Most machine learning algorithms assume that the underlying class distribution (i.e. percentage of examples belonging to each class) is balanced. However, in many real-world applications (e.g. anomaly detection, medical diagnosis, predictive maintenance, driver behavior detection or detection of oil spills), the number of examples from negative class (majority class) significantly outnumbers the number of positive class (minority class or class of interest) examples. In such situations, the traditional machine learning algorithms tend to have bias towards the majority class. This problem of machine learning is known as imbalanced learning or learning from imbalanced data [1].

Related Work. In the literature, many studies have been conducted to address the problem of imbalanced learning. Most of the proposed approaches can be categorized into 3 groups depending on the way they deal with class imbalance. Data level approaches [2]–[6] focus on balancing the input data distribution in order to reduce the effect of class imbalance during the learning process. The algorithm level [1], [7]–[10] approaches focus on developing or modifying the existing algorithms to handle imbalanced datasets by giving more significance to positive examples. Finally, the cost-sensitive approaches [11]–[14] deals with class imbalance by incorporating different classification costs for each class.

Among these approaches, a group of techniques make use of ensembles of classifiers. Ensemble learning [15], [16], aims at combining a set of classifiers in order to build a more efficient classifier than each of the individual classifiers alone.

This strategy has shown to be effective in a large number of applications [17], [18]. While dealing with imbalanced data, one of the main advantages of ensemble learning approaches is that they are versatile to the choice of base learning algorithm. Many ensemble learning based approaches have been proposed to deal with imbalanced datasets, including but not limited to EasyEnsemble [6], SMOTEBagging [19], Balanced Random Forest [20] or SMOTEBoost [21]. In the ensemble learning literature, it is well known that controlling the trade-off between accuracy and diversity among classifiers plays a key role while learning a combination of classifiers [22], [23]. Moreover, *Díez-Pastor et. al.* [24] and *Yao et. al.* [19] showed that approaches that control the diversity among classifiers improves the performance of imbalanced classification tasks. With this in mind, our objective is to design an algorithm for imbalanced datasets which explicitly controls this trade-off between accuracy and diversity among classifiers.

Contribution. In this work, we propose an ensemble method that outputs a **D**iversity-**A**ware weighted **M**ajority **V**ote over previously learned base classifiers for **I**mbalanced datasets (referred to as `DAMVI`). In order to learn weights over the base classifiers, we minimize the upper bound on the error of the majority vote, using PAC-Bayesian C-Bound [25], [26], which allows us to control the trade-off between accuracy and diversity. Concretely, after learning base classifiers for different bootstrapped samples of input data, the algorithm i) increases the weights of positive examples (minority class) which are "hard" to classify with uniformly weighted base classifiers; and ii) then learns weights over base classifiers by optimizing the C-Bound (with focus on "hard"positive examples). The key benefits of our approach are that it does not make any prior assumption on underlying data distribution and it is independent of base learning algorithm. To show the potential of our algorithm, we empirically evaluate our approach on predictive maintenance tasks, credit card fraud detection, webpage classification and medical applications. From our experiments, we show that `DAMVI` is more "consistent"and "stable" compared to state-of-art methods both in terms of F1-measure and Average Precision (AP), in case when we have high imbalance in class distribution ($< 4\%$ of Imbalance Ratio). This is due to the fact that our method is able to explicitly control the trade-off between accuracy and diversity among

classifiers on hard positive examples.

Paper Organization. In the next section, we present general notations and setting for our algorithm. In Section III, we derive our algorithm `DAMVI` for imbalanced datasets. Before concluding in Section V, we present obtained experimental results using our approach in Section IV.

II. NOTATIONS AND SETTING

In this work, we consider a binary classification task where the examples are drawn from a fixed yet unknown distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ is the d-dimensional input space and $\mathcal{Y} = \{-1, +1\}$ the label/output space. Typically, in case of learning with imbalanced data, the percentage of examples belonging to one class is significantly smaller than the another class. In our case, we assume that examples belonging to positive class are in minority. A learning algorithm is provided with training sample of n examples denoted by $S = \{(x_i, y_i)\}_{i=1}^n$, that is assumed to be independently and identically distributed (*i.i.d.*) according to $\mathcal{D}$. We further assume that we have a set of classifiers $\mathcal{H}$ from $\mathcal{X}$ to $\mathcal{Y}$. Given S, our objective is to learn a weight distribution Q over $\mathcal{H}$ that leads to a well performing weighted majority vote (B_Q), such that B_Q

$$B_Q(x) = \text{sign}\left[\mathop{\mathbb{E}}_{h \sim Q} h(x)\right] \quad (1)$$

has the smallest possible generalization error on $\mathcal{D}$ which is highly imbalanced in terms of class distribution. In other words, our goal is to learn Q over $\mathcal{H}$ such that it minimizes the true risk $R_{\mathcal{D}}(B_Q)$ of $B_Q(x)$:

$$R_{\mathcal{D}}(B_Q) = \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \text{I}\left[B_Q(x) \neq y\right] \quad (2)$$

where, $\text{I}[p]$ is equal to 1 if the predicate p is true, and 0 otherwise. An important behavior of the above risk on Q-weighted majority vote B_Q is that it is closely related to the Gibbs risk $R_{\mathcal{D}}(G_Q)$ which is defined as the expectation of the individual risks of each classifier that appears in the majority vote. Formally, we can define Gibbs risk as follows:

$$R_{\mathcal{D}}(G_Q) = \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \mathop{\mathbb{E}}_{h \sim Q} \text{I}[h(x) \neq y].$$

In fact, if B_Q misclassifies $x \in \mathcal{X}$, then at least half of the classifiers (under measure Q) makes a prediction error on x. Therefore. we have

$$R_{\mathcal{D}}(B_Q) \leq 2R_{\mathcal{D}}(G_Q)$$

Thus, an upper bound on $R_{\mathcal{D}}(G_Q)$ gives rise to an upper bound on $R_{\mathcal{D}}(B_Q)$. There exist other tighter relations [25]–[27], such as PAC-Bayesian C-Bound [25] that involves the *expected disagreement* $d_{\mathcal{D}}(Q)$ between pair of classifiers, and that can be expressed as follows (when $R_{\mathcal{D}}(G_Q) \leq \frac{1}{2}$):

$$R_{\mathcal{D}}(B_Q) \leq 1 - \frac{(1 - 2R_{\mathcal{D}}(G_Q))^2}{1 - 2d_{\mathcal{D}}(Q)} \quad (3)$$

$$\text{where } d_{\mathcal{D}}(Q) = \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \mathop{\mathbb{E}}_{h \sim Q} \mathop{\mathbb{E}}_{h' \sim Q} \text{I}[h(x) \neq h'(x)]$$

We provide the proof of above C-Bound in Appendix VI-A. The expected disagreement $d_{\mathcal{D}}(Q)$ measures the diversity/disagreement among classifiers. It is worth noting that from imbalanced data classification standpoint where the notion of diversity among classifiers is known to be important ([19], [24]), Equation 3 directly captures the trade-off between the accuracy and the diversity among classifiers. Therefore, in this work, we propose a new algorithm (presented in next Section III) for imbalanced learning which directly exploits PAC-Bayesian C-Bound in order to learn a weighted majority vote classifier. Note that, the PAC-Bayesian C-Bound has been shown to be an effective approach to learn a weighted majority vote over a set of classifiers in many applications, e.g. multimedia analysis [28] and multiview learning [29], [30].

III. LEARNING A MAJORITY VOTE FOR IMBALANCED DATA

Our objective is to learn weights over a set of classifiers that leads to a well-performing weighted majority vote (given by Equation 1) to deal with imbalanced datasets. It has been shown that controlling the trade-off between accuracy and diversity between the set of classifiers plays an important role for imbalanced classification problems [19], [24]. Therefore, we utilize PAC-Bayesian C-Bound (given by Equation 3) which explicitly controls this trade-off in order to derive a diversity-aware ensemble learning based algorithm (referred as `DAMVI`, see Algorithm 1) for binary imbalanced classification tasks.

For a given training set $S = \{(x_1, y_1), \ldots, (x_n, y_n)\} \in (\mathbb{R}^d \times \{-1, +1\})^n$ of size n; `DAMVI` (Algorithm 1) trains a set of base classifiers $\mathcal{H} = \{h_1, \ldots, h_K\}$ (using a base learning algorithm A) corresponding to K bootstrapped samples (Step 1 to 4) [1]. Then, we propose to update the weights of those training examples which belong to the minority class (in our case , $y_i = 1$) as follows (Step 7):

$$\forall x_i \in S, \ \mathcal{D}(x_i) = \begin{cases} \frac{\mathcal{D}(x_i) \exp\left(-y_i \sum_{k=1}^K Q(h_k) h_k(x_i)\right)}{Z}, & \text{if } y_i = 1 \\ \frac{\mathcal{D}(x_i)}{Z}, & \text{if } y_i = -1 \end{cases}$$

where, $Z = \sum_{j=1}^n \mathcal{D}(x_j)$ is a normalization factor.

In Step 7, the weights of misclassified (resp. correctly classified) positive examples according to the uniformly weighted majority vote classifier increase (resp. decrease). Note that, here we update the weights over the learning sample S just once by focusing only on positive examples. Whereas, boosting algorithms [31] (e.g. Adaboost [32]) repeatedly learn a "weak" classifier using a learning algorithm with different probability distributions over S. Intuitively, this step increases the weights of those positive examples which are "hard"to classify with the uniformly weighted classifier ensemble. This step allows us to focus on "hard" positive examples while learning weights over the base classifiers.

Then, we propose to learn the weights over the classifiers by optimizing the C-Bound on weighted training sample S,

---

[1]Our algorithm is not limited to base learners learnt using bootstrapped samples. It is applicable to any set of base learners.

Algorithm 1 `DAMVI`
***

Input: Training set $S = \{(x_i, y_i), \ldots, (x_n, y_n)\}$, where $x_i \subseteq \mathbb{R}^d$ and $y_i \in \{-1, +1\}$,
 Number of classifiers K
 Base Learning algorithm A.

Initialize: Empty set of classifiers $\mathcal{H} = \{\phi\}$.
$$\forall \, x_i \in S, \mathcal{D}(x_i) \leftarrow \tfrac{1}{n}$$

1: **for** k = 1, ..., K **do**
2:   Generate a bootstrap sample $S(k)$ from S
3:   Learn a classifier h_k using the base learning algorithm A
4:   Update $\mathcal{H} = \mathcal{H} \cup \{h_k\}$
5: **for** $h_k \in \mathcal{H}$ **do**
6:   $Q(h_k) \leftarrow \tfrac{1}{K}$
7: Update the distribution $\mathcal{D}$ over the learning sample S

$$\forall x_i \in S, \mathcal{D}(x_i) = \begin{cases} \dfrac{\mathcal{D}(x_i)\exp\left(-y_i\sum_{k=1}^K Q(h_k)h_k(x_i)\right)}{Z}, & \text{if } y_i = 1 \\ \dfrac{\mathcal{D}(x_i)}{Z}, & \text{if } y_i = -1 \end{cases}$$

where, $Z = \sum_{j=1}^{n} \mathcal{D}(x_j)$ is a normalization factor.

8: **Optimize** the C-Bound to learn weights over classifiers

$$\max_Q \left(\left[1 - 2\sum_{i=1}^n \mathcal{D}(x_i)\sum_{k=1}^K Q(h_k)\mathrm{I}\left[h_k(x_i)\neq y_i\right]\right]^2 \middle/ \right.$$
$$\left[1 - 2\sum_{i=1}^n \mathcal{D}(x_i)\sum_{k=1}^K\sum_{k'=1}^K Q(h_k)Q(h_{k'})\mathrm{I}\left[h_k(x_i)\neq h_{k'}(x_i)\right]\right]\right)$$
$$\text{s.t.} \quad \sum_{k=1}^K Q(h_k) = 1, \; Q(h_k) \geq 0 \;\; \forall k \in \{1, \ldots, K\}$$

9: **Return:** Set of classifiers $\mathcal{H}$ and learned weights over classifiers i.e. Q. Such that, for any input example x, final weighted majority is defined as:

$$B_Q(x) = \operatorname{sign}\left(\sum_{k=1}^K Q(h_k)h_k(x)\right)$$
***

given by Equation 3 (Step 8), which can be represented by the following constraint optimization problem:

$$\max_Q \left(\left[1 - 2\sum_{i=1}^n \mathcal{D}(x_i)\sum_{k=1}^K Q(h_k)\mathrm{I}\left[h_k(x_i)\neq y_i\right]\right]^2 \middle/ \right.$$
$$\left[1 - 2\sum_{i=1}^n \mathcal{D}(x_i)\sum_{k=1}^K\sum_{k'=1}^K Q(h_k)Q(h_{k'})\mathrm{I}\left[h_k(x_i)\neq h_{k'}(x_i)\right]\right]\right)$$
$$\text{s.t.} \quad \sum_{k=1}^K Q(h_k) = 1, \; Q(h_k) \geq 0 \;\; \forall k \in \{1, \ldots, K\}$$

Intuitively, on "hard" positive examples, the C-Bound tries to diversify the classifiers and at the same time controls the classification error of the classifiers which is a key element

for imbalanced datasets [19], [24]. As the above optimization problem is a constrained nonlinear problem, therefore we use a Sequential Quadratic Programming [33] algorithm which uses the quasi-Newton method to find maxima of above optimization problem.

Finally, the learned weights over the classifiers leads to a well-performing majority vote, given by Equation 1, tailored for imbalanced classification tasks. For any input example x, the final learned weighted majority vote is given as follows:

$$B_Q(x) = \operatorname{sign}\left(\sum_{k=1}^K Q(h_k)h_k(x)\right)$$

IV. EXPERIMENTS

In this section, we present an empirical study to show the performance of our algorithm `DAMVI` on the following datasets.

TABLE I: Summary of Datasets: Number of attributes, Number of examples and the Imbalance Ratio (IR) i.e. percentage of positive examples (minority class).

	#Attributes	#Examples	IR
Webpage	300	34780	3.03
Mammography	6	11183	2.32
Scania	170	60000	1.67
Protein Homo	74	145751	0.9
Credit Fraud	30	284807	0.17
PCT Data	17	816099	0.02

A. Datasets

We have validated `DAMVI` on 6 datasets belonging to predictive maintenance task, credit card fraud detection, webpage classification and medical applications. A description of these datasets is presented in Table I.

- **Predictive maintenance** relies on equipment data (telemetry data) and historical maintenance data to track the performance of equipment in order to predict possible failures in advance. We considered a real-world `Scania` dataset [34][2] which is openly available and collected from heavy Scania trucks in everyday usage. The positive class (minority class) corresponds to failures of specific component of the Air Pressure System (APS) and negative class corresponds to failures of components not related to the APS system. The `PCT Data` consists of equipment data (sensor data) and maintenance data from trucks operating at Piraeus Container Terminal (PCT) in Athens, Greece. The positive class (minority class) corresponds to truck failures and the negative class corresponds to normally functioning trucks. This dataset is proprietary and was obtained thanks to a research collaboration.
- **Credit Card Fraud Detection** composed of credit card transactions where positive class (minority class) examples are fraudulent transactions and negative class examples are non-fraudulent. The `Credit Fraud` dataset [35][3] is

***
[2] https://archive.ics.uci.edu/ml/datasets/APS+Failure+at+Scania+Trucks
[3] https://www.kaggle.com/mlg-ulb/creditcardfraud

an openly available real-world dataset consisting of credit card transactions occurred during two days in September, 2013. This dataset was collected and analyzed during a research collaboration between Worldline and ULB (Université Libre de Bruxelles).

- **Medical Datasets:** We considered 2 openly available datasets related to medical applications: `Mammography` and `Protein Homo`[4]. The `Mammography` dataset [36] composed of results from an eponymous breast screening method. The positive class (minority class) corresponds to a malignant mass and the negative class corresponds to a benign mass. The `Protein Homo` dataset [37] is an openly available dataset from 2004 KDD-Cup competition[4]. It is a protein homology prediction task where homologous (*resp.* non-homologous) sequences correspond to the positive (*resp.* negative) class.

- `Webpage` [36] is an openly available text classification dataset[4] where the objective is to identify whether a webpage belongs to a particular category (positive class) or not.

B. Experimental Protocol

To study the performance of `DAMVI`, we considered following 9 baseline approaches [8]:

- **Random Oversampling + Decision Tree(`R-DT`):** This approach first balances the class distribution by randomly replicating minority class examples. Then, we learn a decision tree classifier on oversampled data.

- **SMOTE + Decision Tree (`S-DT`):** This approach first oversamples the minority class examples using Synthetic Minority Over Sampling Technique (SMOTE) [3] algorithm. SMOTE oversamples the minority class examples by interpolating between several minority class examples that lie together. After oversampling, we learn a decision tree classifier.

- **ADASYN + Decision Tree (`A-DT`):** This approach first oversamples the minority class examples using Adaptive Synthetic (ADASYN) sampling algorithm [4]. ADASYN computes a weight distribution over minority class examples to synthetically generate data for minority class examples that are harder to learn. After oversampling, we learn a decision tree classifier.

- **ROSBagging (`R-BG`)** [8]: This approach first oversamples the minority class examples by following Random Oversampling (ROS) approach. Then, we learn an ensemble of decision tree classifiers on bootstrapped samples of oversampled data.

- **SMOTEBagging (`S-BG`)** [21]:This approach first oversamples the minority class examples following SMOTE algorithm. Then, we learn an ensemble of decision tree classifiers on bootstrapped samples of oversampled data.

- **ADASYNBagging (`A-BG`):**This approach first oversamples the minority class examples following ADASYN

algorithm. Then, we learn an ensemble of decision tree classifiers on bootstrapped samples of oversampled data.

- **Balanced Bagging (`BB`)** [8]: This approach balances the dataset using random undersampling. Then, an ensemble of decision tree classifiers are learnt on bootstrapped samples of oversampled data.

- **Balanced Random Forest (`BRF`)** [20]: This approach learns an ensemble of classification trees from balanced bootstrapped samples of original input data.

- **Easy Ensemble (`EE`)** [6]: This approach learns an ensemble of AdaBoost learners trained on different balanced bootstrap samples.

For all oversampling based approaches (`R-DT, S-DT, A-DT, R-BG, S-BG, A-BG`), we used the ROS, SMOTE and ADASYN implementations of `imbalanced-learn` Python package [36] to synthetically generate new minority class examples such that the number of minority class examples is equal to the number of majority class examples. For SMOTE and ADASYN, we considered 5 nearest neighbours to generate synthetic examples.

For `EE, BB` and `BRF`, we used implementations of `imbalanced-learn` Python package where the number of base learners is equal to 100. For our approach `DAMVI`[5] and baselines `R-BG, S-BG, A-BG`, we fix the number of decision tree classifiers to 100 and size of bootstrapped sample to 20% of the size of original training data. For our approach `DAMVI`, we learn the weights over base classifiers by optimizing C-Bound on weighted training sample S. For solving the constrained optimization problem, we used Sequential Least SQuares Programming (SLSQP) implementation of scikit-learn [38] (that we also used to learn the decision tree classifiers) with uniform initialization of weights over the base classifiers. For all the experiments, we reserved 30% of data for testing and the remaining for training. Experiments are repeated 5 times by each time splitting the training and the test sets at random over the initial datasets.

Evaluation Metrics: Under the imbalanced learning scenario, the conventional evaluation metrics such as accuracy are unable to adequately represent the model's performance on the minority class examples which is typically the class of interest [1]. Therefore, we evaluate the models based on two metrics: F1-score and Average Precision (AP), which are known to be relevant for imbalanced classification problems [1], [8], [39]. F1-score is defined as harmonic mean of precision and recall. Whereas, Average Precision (AP) is the area under the precision-recall curve and it has been shown that AP, in case of highly imbalanced datasets, is more informative than AUC ROC [39].

C. Results

Firstly, we report the comparison of our algorithm `DAMVI` with all the considered baselines in Table II (for F1-score) and Table III (for Average Precision). As shown in Tables II and

---

[4]https://imbalanced-learn.readthedocs.io/en/stable/generated/imblearn. datasets.fetch_datasets.html

[5]`DAMVI` codes are available at https://github.com/goyalanil/DAMVI

TABLE II: F1-score for different approaches averaged over 5 random sets. Along each column, the best result is in bold, and second one in italic. $\downarrow$ indicates that a result is statistically significantly worse than DAMVI, according to Wilcoxon rank sum test [40] with $p < 0.05$

	Webpage	Mammography	Scania	Protein Homo	Credit Fraud	PCT Data
S-DT	.5062±.028$\downarrow$	.5198±.009$\downarrow$	.5848±.011$\downarrow$	.5278±.005$\downarrow$	.5610±.016$\downarrow$	.8793±.011$\downarrow$
R-DT	.4705±.021$\downarrow$	.6053±.043$\downarrow$	.6256±.019$\downarrow$	.7290±.017$\downarrow$	.7556±.013$\downarrow$	.9715±.002$\downarrow$
A-DT	.4693±.019$\downarrow$	.4978±.034$\downarrow$	.5807±.020$\downarrow$	.5259±.019$\downarrow$	.5653±.027$\downarrow$	.8830±.009$\downarrow$
R-BG	.4620±.016$\downarrow$	*.6145±.026*$\downarrow$	*.6845±.014*$\downarrow$	.7849±.021$\downarrow$	*.7703±.020*$\downarrow$	.9691±.001$\downarrow$
S-BG	*.6134±.017*$\downarrow$	.5391±.017$\downarrow$	.6493±.009$\downarrow$	.6771±.009$\downarrow$	.6839±.024$\downarrow$	.9430±.006$\downarrow$
A-BG	.4804±.021$\downarrow$	.5169±.011$\downarrow$	.6269±.007$\downarrow$	.6346±.013$\downarrow$	.6819±.030$\downarrow$	.9312±.004$\downarrow$
BB	.3445±.001$\downarrow$	.4465±.030$\downarrow$	.4317±.005$\downarrow$	.4275±.008$\downarrow$	.1376±.006$\downarrow$	.8014±.006$\downarrow$
BRF	.4098±.010$\downarrow$	.3659±.014$\downarrow$	.3822±.004$\downarrow$	.4027±.009$\downarrow$	.1255±.016$\downarrow$	.2943±.007$\downarrow$
EE	.4678±.011$\downarrow$	.2534±.002$\downarrow$	.4096±.006$\downarrow$	.3350±.003$\downarrow$	.0922±.007$\downarrow$	.0881±.001$\downarrow$
DAMVI	**.7996**±.011	**.6661**±.023	**.7289**±.011	**.8067**±.009	**.8495**±.019	**.9816**±.001

TABLE III: Average Precision (AP) for different approaches averaged over 5 random sets. Along each column, the best result is in bold, and second one in italic. $\downarrow$ indicates that a result is statistically significantly worse than DAMVI, according to Wilcoxon rank sum test [40] with $p < 0.05$

	Webpage	Mammography	Scania	Protein Homo	Credit Fraud	PCT Data
S-DT	.2794±.023$\downarrow$	.2919±.010$\downarrow$	.3526±.014$\downarrow$	.3153±.005$\downarrow$	.3482±.016$\downarrow$	.7785±.001$\downarrow$
R-DT	.3008±.016$\downarrow$	.3811±.054$\downarrow$	.3994±.024$\downarrow$	.5347±.025$\downarrow$	.5728±.020$\downarrow$	.9447±.005$\downarrow$
A-DT	.2481±.014$\downarrow$	.2740±.034$\downarrow$	.3483±.023$\downarrow$	.3112±.019$\downarrow$	.3516±.030$\downarrow$	.7851±.016$\downarrow$
R-BG	.4944±.010$\downarrow$	*.7011*±.021	*.8097±.016*$\downarrow$	.8495±.016	.8120±.030$\downarrow$	*.9875*±.001
S-BG	.6219±.028$\downarrow$	.6971±.025$\downarrow$	.7275±.019$\downarrow$	.8424±.013	.8135±.027$\downarrow$	.9863±.001$\downarrow$
A-BG	.4400±.024$\downarrow$	.6261±.036$\downarrow$	.6712±.018$\downarrow$	.8276±.016	*.8137±.035*$\downarrow$	.9847±.005$\downarrow$
BB	.6302±.034$\downarrow$	.6644±.037$\downarrow$	.6745±.024$\downarrow$	.8359±.018	.7516±.048$\downarrow$	.9849±.001$\downarrow$
BRF	.6930±.022$\downarrow$	.6782±.023	.6877±.016$\downarrow$	*.8549*±.014	.7615±.047$\downarrow$	.6976±.010$\downarrow$
EE	.6969±.038$\downarrow$	.5967±.043$\downarrow$	.7558±.014$\downarrow$	**.8561**±.012	.7672±.025$\downarrow$	.0790±.001$\downarrow$
DAMVI	**.8331**±.013	**.7142**±.039	**.8335**±.007	.8267±.013	**.8373**±.027	**.9976**±.001

III, our proposed algorithm DAMVI performs best compared to baseline approaches for all datasets in terms of F1-score and for 5 out of 6 datasets in terms of Average Precision. Moreover, on PCT Data (where we have lowest imbalance ratio i.e. 0.02), we perform significantly better than the baselines. According to Wilcoxon rank sum test [40], in most of cases, we are significantly better than the baselines with $p < 0.05$. We can also remark that DAMVI is more "stable" than R-BG (in general, second best approach) according to standard deviation values. Note that R-BG, S-BG, A-BG, EE, BB and BRF are able to create a diverse set of base classifiers on bootstrapped samples of input data. However, these approaches don't focus on learning the weights over the base classifiers tailored for imbalanced datasets. Whereas, DAMVI explicitly learns the weights by controlling the trade-off between the accuracy and the diversity among base classifiers by minimizing PAC-Bayesian C-Bound (with focus on "hard" positive examples). Our results provide evidence that learning a diversity-aware weighted majority vote classifier is an effective way to deal with imbalanced datasets.

We also analyze the behaviour of all the approaches by artificially increasing and decreasing the imbalance for the Mammography dataset. In order to create a dataset with a higher percentage of minority class examples than in the original dataset, we randomly undersample the majority class examples. Similarly, to create a dataset with a lower percentage

of minority class examples than in the original dataset, we randomly undersample the minority class examples. Figure 1 illustrates the obtained results by showing the evolution of F1-score and Average Precision with respect to the imbalance ratio (i.e. percentage of positive class examples) on the Mammography dataset. As shown in Figure 1, DAMVI performs better than baselines both in terms of F1-score and AP when the imbalance ratio (IR) is less than 4% (except at 2% for F1-score). This shows that DAMVI performs well even for highly imbalanced classification tasks ($< 4\%$ of IR). Below 1% of IR, we can notice that EasyEnsemble (EE) gradually performs second best in terms of AP (but worst in terms of F1-score) and ROSBagging (R-BG) performs second best in terms of F1-score (but drastically drops in terms of AP). However, our approach DAMVI remains "consistent" and "stable" both in terms of F1-score and AP throughout the evolution of imbalance ratio. This shows that explicitly controlling the trade-off between the accuracy and the diversity among classifiers (by focusing on "hard"positive examples) plays an important role while learning an ensemble of classifiers for imbalanced datasets.

A note on the Complexity of the Algorithm: The complexity of learning a decision tree classifier is $O(d.n.\log(n))$, where d is the dimension of input space. We learn the weights over the base classifiers by optimizing Equation (3 (Step 8 of our algorithm) using SLSQP method which has time complexity
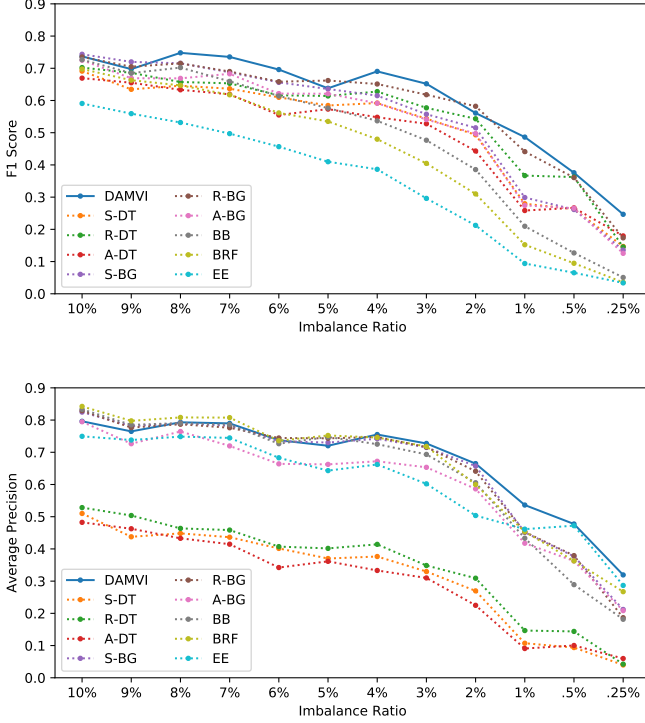
Fig. 1: Evolution of F1-score and Average Precision *w.r.t* Imbalance Ratio on `Mammography` dataset.

of $O(K^3)$. Therefore, the overall complexity of `DAMVI` is $O(K.d.n. \log(n) + K^3)$. Note that we can easily parallelize `DAMVI`: by using K machines, we can learn decision tree classifiers parallelly and weights over them.

V. CONCLUSION

In this paper, we considered the problem of imbalanced learning where the number of negative examples (majority class) significantly outnumbers the positive class (minority class or class of interest) examples. In order to deal with imbalanced datasets, we propose an ensemble learning based algorithm (referred to as `DAMVI`) that learns a diversity-aware weighted majority vote classifier over the base classifiers. After learning base classifiers, the algorithm i) increases the weights of positive examples (minority class) which are "hard" to classify with uniformly weighted base classifiers; and ii) then learns weights over base classifiers by optimizing the PAC-Bayesian C-Bound. We have validated our approach on various datasets and we show that `DAMVI` consistently performs better than state-of-art models. We also show that explicitly controlling the trade-off between the accuracy and the diversity among base classifiers (with focus on hard positive examples) is an effective strategy to deal with highly imbalanced datasets.

As future work, we would like to extend our algorithm to the *semi-supervised case*, where one has access to an additionally unlabeled set during the training. One possible way is to learn base classifiers using pseudo-labels (for unlabeled data)

generated from the K-means classifier trained using labeled data. We would also like to extend our algorithm to the case of multiclass imbalanced classification problems. One possible solution is to make use of multiclass C-Bound [41] to learn the diversity-aware weighted majority vote classifier.

VI. APPENDIX

A. Proof of C-Bound

In this section, we present the proof of C-Bound (Equation 3), similar to the proof provided by Germain *et al.* [26]. Firstly, we need to define the margin of the weighted majority vote B_Q and its first and second statistical moments.

Definition VI.1. *Let M_Q is a random variable that outputs the margin of the weighted majority vote on the example (x, y) drawn from distribution $\mathcal{D}$, given by:*

$$M_Q(x, y) = \mathop{\mathbb{E}}_{h \sim Q} y\, h(x).$$

The first and second statistical moments of the margin are respectively given by

$$\mu_1(M_Q^{\mathcal{D}}) = \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} M_Q(x, y). \tag{4}$$

and,

$$
\begin{aligned}
\mu_2(M_Q^{\mathcal{D}}) &= \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \big[M_Q(x, y) \big]^2 \\
&= \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} y^2 \Big[\mathop{\mathbb{E}}_{h \sim Q} h(x^v) \Big]^2 = \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \Big[\mathop{\mathbb{E}}_{h \sim Q} h(x) \Big]^2.
\end{aligned}
\tag{5}
$$

According to this definition, the risk of the weighted majority vote can be rewritten as follows:

$$R_{\mathcal{D}}(B_Q) = \mathop{\Pr}_{(x,y) \sim \mathcal{D}} \big(M_Q(x, y) \leq 0 \big).$$

Moreover, the risk of the Gibbs classifier can be expressed thanks to the first statistical moment of the margin. Note that in the binary setting where $y \in \{-1, 1\}$ and $h : \mathcal{X} \to \{-1, 1\}$, we have $\mathrm{I}[h(x) \neq y] = \frac{1}{2}(1 - y\, h(x))$, and therefore

$$
\begin{aligned}
R_{\mathcal{D}}(G_Q) &= \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \mathop{\mathbb{E}}_{h \sim Q} \mathrm{I}[h(x) \neq y] \\
&= \frac{1}{2} \left(1 - \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \mathop{\mathbb{E}}_{h \sim Q} y\, h(x) \right) \\
&= \frac{1}{2}(1 - \mu_1(M_Q^{\mathcal{D}})).
\end{aligned}
\tag{6}
$$

Similarly, the expected disagreement can be expressed thanks to the second statistical moment of the margin by

$$
\begin{aligned}
d_{\mathcal{D}}(Q) &= \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \mathop{\mathbb{E}}_{h \sim Q} \mathop{\mathbb{E}}_{h' \sim Q} \mathrm{I}[h(x) \neq h'(x)] \\
&= \frac{1}{2} \left(1 - \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \mathop{\mathbb{E}}_{h \sim Q} \mathop{\mathbb{E}}_{h' \sim Q} h(x) \times h'(x) \right) \\
&= \frac{1}{2} \left(1 - \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \Big[\mathop{\mathbb{E}}_{h \sim Q} h(x) \Big] \times \Big[\mathop{\mathbb{E}}_{h' \sim Q} h'(x) \Big] \right) \\
&= \frac{1}{2} \left(1 - \mathop{\mathbb{E}}_{(x,y) \sim \mathcal{D}} \Big[\mathop{\mathbb{E}}_{h \sim Q} h(x) \Big]^2 \right) \\
&= \frac{1}{2}(1 - \mu_2(M_Q^{\mathcal{D}})).
\end{aligned}
\tag{7}
$$

From above, we can easily deduce that $0 \leq d_{\mathcal{D}}(Q) \leq 1/2$ as $0 \leq \mu_2(M_Q^{\mathcal{D}}) \leq 1$. Therefore, the variance of the margin can be written as:

$$\begin{aligned}
\mathrm{Var}(M_Q^{\mathcal{D}}) &= \underset{(x,y)\sim\mathcal{D}}{\mathbf{Var}}(M_Q(x,y)) \\
&= \mu_2(M_Q^{\mathcal{D}}) - (\mu_1(M_Q^{\mathcal{D}}))^2.
\end{aligned} \tag{8}$$

The proof of the $\mathcal{C}$-bound

Proof. By making use of one-sided Chebyshev inequality (Theorem 1 in Appendix VI-B), with $X = -M_Q(x,y)$, $\mu = \underset{(x,y)\sim\mathcal{D}}{\mathbb{E}}(M_Q(x,y))$ and $a = \underset{(x,y)\sim\mathcal{D}}{\mathbb{E}} M_Q(x,y)$, we have

$$R_{\mathcal{D}}(B_Q) = \underset{(x,y)\sim\mathcal{D}}{\mathrm{Pr}}\left(M_Q(x,y) \leq 0\right)$$

$$= \underset{(x,y)\sim\mathcal{D}}{\mathrm{Pr}}\left(-M_Q(x,y) + \underset{(x,y)\sim\mathcal{D}}{\mathbb{E}} M_Q(x,y) \geq \underset{(x,y)\sim\mathcal{D}}{\mathbb{E}} M_Q(x,y)\right)$$

$$\leq \frac{\underset{(x,y)\sim\mathcal{D}}{\mathbf{Var}}(M_Q(x,y))}{\underset{(x,y)\sim\mathcal{D}}{\mathbf{Var}}(M_Q(x,y)) + \left(\underset{(x,y)\sim\mathcal{D}}{\mathbb{E}} M_Q(x,y)\right)^2}$$

$$= \frac{\mathrm{Var}(M_Q^{\mathcal{D}})}{\mu_2(M_Q^{\mathcal{D}}) - \left(\mu_1(M_Q^{\mathcal{D}})\right)^2 + \left(\mu_1(M_Q^{\mathcal{D}})\right)^2}$$

$$= \frac{\mathrm{Var}(M_Q^{\mathcal{D}})}{\mu_2(M_Q^{\mathcal{D}})}$$

$$= \frac{\mu_2(M_Q^{\mathcal{D}}) - \left(\mu_1(M_Q^{\mathcal{D}})\right)^2}{\mu_2(M_Q^{\mathcal{D}})}$$

$$= 1 - \frac{\left(\mu_1(M_Q^{\mathcal{D}})\right)^2}{\mu_2(M_Q^{\mathcal{D}})}$$

$$= 1 - \frac{\left(1 - 2 R_{\mathcal{D}}(G_Q)\right)^2}{1 - 2 d_{\mathcal{D}}(Q)}$$

$\square$

B. Mathematical Tools

Theorem 1 (Cantelli-Chebyshev inequality)**.** *For any random variable X s.t. $\mathbb{E}(X) = \mu$ and $\mathbf{Var}(X) = \sigma^2$, and for any $a > 0$, we have $\mathbb{P}(X - \mu \geq a) \leq \frac{\sigma^2}{\sigma^2 + a^2}$.*

REFERENCES

[1] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[2] G. E. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD explorations newsletter*, vol. 6, no. 1, pp. 20–29, 2004.

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

[4] H. He, Y. Bai, E. A. Garcia, and S. Li, "Adasyn: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*. IEEE, 2008, pp. 1322–1328.

[5] A. Liu, J. Ghosh, and C. E. Martin, "Generative oversampling for mining imbalanced datasets." in *DMIN*, 2007, pp. 66–72.

[6] X.-Y. Liu, J. Wu, and Z.-H. Zhou, "Exploratory undersampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp. 539–550, 2008.

[7] W. Fan, S. J. Stolfo, J. Zhang, and P. K. Chan, "Adacost: misclassification cost-sensitive boosting," in *Icml*, vol. 99, 1999, pp. 97–105.

[8] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463–484, 2011.

[9] B. Zadrozny, J. Langford, and N. Abe, "Cost-sensitive learning by cost-proportionate example weighting." in *ICDM*, vol. 3, 2003, p. 435.

[10] G. Wu and E. Y. Chang, "Kba: Kernel boundary alignment considering imbalanced data distribution," *IEEE Transactions on Knowledge & Data Engineering*, no. 6, 2005.

[11] C. Elkan, "The foundations of cost-sensitive learning," in *International joint conference on artificial intelligence*, vol. 17, no. 1. Lawrence Erlbaum Associates Ltd, 2001, pp. 973–978.

[12] N. Thai-Nghe, Z. Gantner, and L. Schmidt-Thieme, "Cost-sensitive learning methods for imbalanced data," in *The 2010 International joint conference on neural networks (IJCNN)*. IEEE, 2010, pp. 1–8.

[13] C. X. Ling and V. S. Sheng, "Cost-sensitive learning and the class imbalance problem," pp. 231–235, 2008.

[14] X. Yuan, L. Xie, and M. Abouelenien, "A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data," *Pattern Recognition*, vol. 77, pp. 160–172, 2018.

[15] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*, 2000, pp. 1–15.

[16] M. Re and G. Valentini, "Ensemble methods: a review," *Advances in machine learning and data mining for astronomy*, pp. 563–582, 2012.

[17] N. C. Oza and K. Tumer, "Classifier ensembles: Select real-world applications," *Information Fusion*, vol. 9, no. 1, pp. 4–20, 2008.

[18] C. Zhang and Y. Ma, *Ensemble machine learning: methods and applications*. Springer, 2012.

[19] S. Wang and X. Yao, "Diversity analysis on imbalanced data sets by using ensemble models," in *2009 IEEE Symposium on Computational Intelligence and Data Mining*. IEEE, 2009, pp. 324–331.

[20] T. M. Khoshgoftaar, M. Golawala, and J. Van Hulse, "An empirical study of learning from imbalanced data using random forest," in *19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2007)*, vol. 2. IEEE, 2007, pp. 310–317.

[21] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "Smoteboost: Improving prediction of the minority class in boosting," in *European conference on principles of data mining and knowledge discovery*. Springer, 2003, pp. 107–119.

[22] G. Brown and L. I. Kuncheva, ""good" and "bad" diversity in majority vote ensembles," in *Multiple Classifier Systems*, N. El Gayar, J. Kittler, and F. Roli, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 124–133.

[23] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*. Wiley-Interscience, 2004.

[24] J. F. Diez-Pastor, J. J. Rodriguez, C. I. García-Osorio, and L. I. Kuncheva, "Diversity techniques improve the performance of the best imbalance learning ensembles," *Inf. Sci.*, vol. 325, no. C, pp. 98–117, Dec. 2015.

[25] A. Lacasse, F. Laviolette, M. Marchand, P. Germain, and N. Usunier, "PAC-Bayes bounds for the risk of the majority vote and the variance of the Gibbs classifier," in *NIPS*, 2006, pp. 769–776.

[26] P. Germain, A. Lacasse, F. Laviolette, M. Marchand, and J. Roy, "Risk bounds for the majority vote: from a PAC-Bayesian analysis to a learning algorithm," *JMLR*, vol. 16, pp. 787–860, 2015.

[27] J. Langford and J. Shawe-Taylor, "PAC-Bayes & margins," in *NIPS*. MIT Press, 2002, pp. 423–430.

[28] E. Morvant, A. Habrard, and S. Ayache, "Majority vote of diverse classifiers for late fusion," in *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*. Springer, 2014, pp. 153–162.

[29] A. Goyal, E. Morvant, P. Germain, and M.-R. Amini, "Pac-bayesian analysis for a two-step hierarchical multiview learning approach," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2017, pp. 205–221.

[30] Anil Goyal, Emilie Morvant, Pascal Germain and Massih-Reza Amini, "Multiview boosting by controlling the diversity and the accuracy of view-specific voters," *Neurocomputing*, vol. 358, pp. 81 – 92, 2019.

[31] Y. Freund, "Boosting a weak learning algorithm by majority," *Inf. Comput.*, vol. 121, no. 2, pp. 256–285, Sep. 1995.

[32] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997. [Online]. Available: http://dx.doi.org/10.1006/jcss.1997.1504

[33] J. Nocedal and S. Wright, *Numerical optimization*. Springer Science & Business Media, 2006.

[34] D. Dua and C. Graff, "UCI machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[35] A. Dal Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating probability with undersampling for unbalanced classification," in *2015 IEEE Symposium Series on Computational Intelligence*. IEEE, 2015, pp. 159–166.

[36] G. Lemaître, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning," *Journal of Machine Learning Research*, vol. 18, no. 17, pp. 1–5, 2017.

[37] R. Caruana, T. Joachims, and L. Backstrom, "Kdd-cup 2004: results and analysis," *ACM SIGKDD Explorations Newsletter*, vol. 6, no. 2, pp. 95–108, 2004.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[39] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 233–240.

[40] E. Lehmann, *Nonparametric Statistical Methods Based on Ranks*. McGraw-Hill, 1975.

[41] F. Laviolette, E. Morvant, L. Ralaivola, and J.-F. Roy, "On generalizing the c-bound to the multiclass and multi-label settings," 2014.