



# On the use of the descriptive variable for enhancing the aggregation of crowdsourced labels

Iker Beñaran-Muñoz<sup>1</sup> · Jerónimo Hernández-González<sup>2</sup> · Aritz Pérez<sup>1</sup>

Received: 25 April 2021 / Revised: 1 August 2022 / Accepted: 6 August 2022  
© The Author(s) 2022

## Abstract

The use of crowdsourcing for annotating data has become a popular and cheap alternative to expert labelling. As a consequence, an aggregation task is required to combine the different labels provided and agree on a single one per example. Most aggregation techniques, including the simple and robust majority voting—to select the label with the largest number of votes—disregard the descriptive information provided by the explanatory variable. In this paper, we propose domain-aware voting, an extension of majority voting which incorporates the descriptive variable and the rest of the instances of the dataset for aggregating the label of every instance. The experimental results with simulated and real-world crowdsourced data suggest that domain-aware voting is a competitive alternative to majority voting, especially when a part of the dataset is unlabelled. We elaborate on practical criteria for the use of domain-aware voting.

**Keywords** Crowdsourcing · Label aggregation · Descriptive variable

## 1 Introduction

In the last decade, the machine learning community has resorted to crowdsourcing for obtaining labelled data at a relatively low cost. Instead of relying on costly experts with low availability for labelling their datasets, crowds of non-expert workers (or annotators), which are available for this type of short tasks, are employed. The main issue is that the expertise of annotators is not guaranteed and their labelling might be misleading. To work around this

---

✉ Jerónimo Hernández-González  
jeronimo.hernandez@ub.edu

Iker Beñaran-Muñoz  
iker.benaran@gmail.com

Aritz Pérez  
aperez@bcamath.org

<sup>1</sup> Basque Center for Applied Mathematics, Bilbao, Spain

<sup>2</sup> Mathematics and Computer Science department, Universitat de Barcelona (UB), Barcelona, Spain

problem, each example is usually labelled by many annotators, assuming that the consensus label is more reliable than each single annotation [1, 2].

The process of inferring the consensus label from a (multi)set of labels is known as *label aggregation*. This process, by the nature of crowdsourcing, is usually focused on being cost-effective, that is, to reach the maximum accuracy counting on the minimum resources. That involves requiring non-expert annotators and as few labels as possible. The simplest yet effective technique is majority voting (MV), where the consensus label is the one with the largest number of votes among annotators for each specific instance. . Many other methods have been proposed but, surprisingly, the descriptive information provided by the explanatory variable of the instances, available in every machine learning problem, is rarely used to enhance label aggregation. Similarly, given an observed instance, the rest of the dataset is usually not taken into account for label aggregation. Our intuition is that useful information for label aggregation can be inferred from other instances through the descriptive variable, assuming that the class conditional distribution evolves smoothly with respect to the instance space.

In this paper, we propose *domain-aware voting (DAV)*, an extension of MV that carries out label aggregation by efficiently combining the labels available for the example at hand and using its explanatory data to gather information from the rest of the instances of the dataset. Thus, it can produce the correct labelling even when an example has never been annotated. A simple way to understand our proposal is to think of the  $k$ -nearest neighbours classifier [3]: it predicts a class distribution based on the neighbours of an example. In our framework, the annotations provided for nearby examples would form the predicted class distribution and this would be added as an extra vote to the label aggregation process. Nevertheless, DAV is a general solution that exploits the domain information by using an estimate of the class conditional distribution which might have been obtained in diverse ways. That domain information is transformed into an extra vote that is obtained at zero cost. In our extensive empirical validation, DAV performs equal or better than MV in most scenarios with fewer labels. That is, DAV outperforms MV in terms of cost-effectiveness and its use can lead to reducing costs of labelling through crowdsourcing, which is the ultimate goal of resorting to a crowd of annotators for labelling.

The rest of the paper is organised as follows: Firstly, the related work is presented, and then, the problem is formally defined and our proposal is presented. In Sect. 5, the hypotheses behind our method are tested through an extensive experimental setting, and the results are broadly discussed next. Finally, we draw the conclusions and suggest open questions as future work.

## 2 Related work

Recently, crowdsourcing has become popular and many platforms, such as Amazon MTurk or Figure-Eight.com, have been born to put in contact workers and task schedulers for data collection, where the use of mobile devices is gaining a central role [4]. Crowdsourcing has been used for a variety of purposes: labelling of data examples for machine learning [5], text correction [6], text translation [7], various forms of disease diagnosis [8], among others [9].

Regarding label aggregation, the robustness of MV, a popular strategy that is explicitly or implicitly used by many methods, has been extensively studied [1, 10]: as long as many annotators take part, they might not be experts. It stands out as it does not model the crowd. Our proposal shares this same level of simplicity. Other methods estimate the reliability

of the annotators, such as weighted voting, which uses the reliability estimates to weigh their votes [11]. Many methods use the Expectation-Maximisation strategy, starting from the seminal work of [12], to iteratively infer a better fit of the model of annotators and improve the estimate of the class labels. It has been combined with spectral methods [13] and deep learning [14], among others. Other methods do not have an explicit voting step and introduce the crowd information into more complex models [15]. The information from the descriptive variable is rarely used for label aggregation. For example, [15–17] used it mainly for estimating the reliability of the annotators. [2] considered it to model the difficulty of the instances within a framework of active learning. [18] use the features and the labels to generate clusters in two layers, that are finally related to the true class labels. In general, the contribution of the descriptive information into the final aggregation is indirect and hardly measurable. Our proposal is a simple voting method that directly integrates the features.

A commonly related task is that of learning from this type of data, known as crowd learning [16]. Crowd learning methodologies can be roughly divided into (i) those that perform label aggregation as the predictive model is learnt [10, 14–16] and (ii) techniques that approach label aggregation and model learning sequentially [2, 11, 12, 17, 18]. This paper would be useful for those in this second group, as we solve the problem of label aggregation, which can then be combined with any classical learning algorithm.

### 3 Problem formalisation and background

The context of this work is a supervised learning problem where a training dataset has been labelled by a crowd of annotators. Formally, let  $X$  and  $C$  be two random variables where  $(X, C)$  is distributed according to  $p(x, c)$ ,  $X$  is the  $d$ -dimensional descriptive variable and  $C$  is the categorical class variable that takes values in the domain  $\Omega = \{1, \dots, r\}$ . A dataset  $D = \{x_i\}_{i=1}^n$  with  $n$  unlabelled instances is provided. The real class label  $c_x$  of the instances  $x$  is hidden, and only a multiset of labels  $S_x = \{1^{m_1}, \dots, r^{m_r}\}$  is available for each  $x \in D$ , where  $m_c \in \mathbb{Z}^+$  is the number of appearances of the class label  $c$  in  $S_x$ . The labels  $l \in S_x$  are provided by the annotators from a crowd. In this work, we assume that annotators (i) provide labels independently and (ii) tend to provide the correct label with the highest probability. That is, neither colluding nor adversarial annotators are considered. Under realistic fair conditions [1],  $S_x$  provides relevant information about the true class  $c_x$  of instance  $x$ . Label aggregation can be formally defined as the procedure of assigning a class  $\hat{c}_x$  to each instance  $x \in D$ , based on the information at hand: the instances and their collections of labels. The goal is to recover the true label of  $x$ , that is, to obtain  $\hat{c}_x = c_x$  as frequently as possible. Let us define a labelling  $L$  as a tuple that assigns a label,  $\hat{c}_x \in \Omega$ , to each instance of the dataset,  $L = \{\hat{c}_x : x \in D\}$ . Thus, the goal of label aggregation can be redefined as to infer a labelling  $L$  that maximises the *aggregation accuracy*:

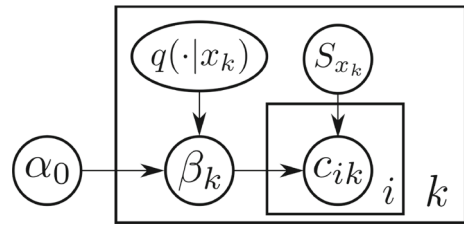
$$a(L) = \frac{1}{n} \sum_{x \in D} \mathbb{1}(\hat{c}_x = c_x) \tag{1}$$

where  $\mathbb{1}(cond) = 1$  if  $cond = true$  and 0 otherwise.

The majority voting (MV) function for  $x$  can be formally written as

$$v(S_x) = \arg \max_{c \in \Omega} v(c|S_x) \tag{2}$$

**Fig. 1** Bayesian interpretation of DAV: For each instance  $x_k$ ,  $\alpha_0$  and  $q(c|x)$  provide the hyperparameters of the Dirichlet prior for that instance,  $\beta_k$ . That Dirichlet prior, along with the collection of labels  $S_{x_k}$ , provides the posterior probabilities for  $c_{ik}$ , for each  $i \in \{1, \dots, r\}$



where  $v(\cdot|S_x)$  is the *voting estimate*, which corresponds to the maximum likelihood estimate of the class:

$$v(c|S_x) = \frac{1}{|S_x|} \sum_{l \in S_x} \mathbf{1}(l = c) \quad (3)$$

## 4 Domain-aware voting

The class uncertainty surrounding an instance decreases as the size of the multiset  $S_x$  increases (more annotations) and the number of distinct labels decreases (annotations concentrated on particular classes, best case single class). In instances with high class uncertainty, the information obtained solely from  $S_x$  may be insufficient. The incorporation of the descriptive information of instances into the voting could enhance the performance of label aggregation. Moreover, we can also incorporate into the aggregation task the intuition that examples with similar descriptive vectors might also share the same class.

In this work, we propose an extension of the classical MV approach which makes use of the explanatory variable  $X$  to incorporate information regarding all the instances from the dataset. When inferring the class label of an instance  $x$ , its descriptive information is exploited along with the information provided by the multiset of labels  $S_x$ . Our proposal, called **domain-aware voting (DAV)**, can be expressed as follows,

$$v^*(x, S_x) = \arg \max_{c \in \Omega} v^*(c|x, S_x) \quad (4)$$

where the DAV estimate,  $v^*(c|x, S_x)$ , is:

$$v^*(c|x, S_x) = \frac{1}{|S_x| + \alpha_0} [|S_x| \cdot v(c|S_x) + \alpha_0 \cdot q(c|x)] \quad (5)$$

The  $q(c|x)$  is an estimate of the conditional class distribution  $p(c|x)$ , which we will call **domain vote**. It is an extra vote added to the voting estimate, which is weighted by means of a parameter  $\alpha_0$ . DAV becomes the MV strategy when  $\alpha_0 = 0$ . Interestingly, the DAV estimate has self-regulatory properties for the aggregation of annotations. Given a fixed  $\alpha_0$  value, the influence of the domain vote in DAV decreases as the size of the collection  $S_x$  increases. In other words, as the number of collected labels tends to infinity, the DAV estimate tends to the voting estimate. Similarly, as the collection  $S_x$  is reduced, the information provided by the domain vote gains relevance. This self-regulatory behaviour is particularly suitable for crowd-labelling scenarios in which the size of the collections of labels of the different instances is typically unbalanced.

In fact, assuming that  $q(c|x)$  is given a priori, DAV can be understood as a Bayesian estimate of the class distribution for an instance (see in Fig. 1 its plate model), where domain votes are the hyperparameters of a Dirichlet prior,  $\beta_k = \alpha_0 \cdot q(c|x_k)$ . In this viewpoint,

$\alpha_0$  is the equivalent sample size, which weighs the contribution of the domain votes to the aggregation scheme.

Note that DAV is a general method where the domain votes can be obtained through a variety of means: They could be considered as priors, the output of a classification model, or obtained through density estimation, to name a few. Throughout the remainder of the paper, we call **domain voter** to a classifier learnt with the voting estimates of the instances as probabilistic ground truth, which introduces the descriptive information into the label aggregation task. A classifier that can provide a probability distribution over the class labels is preferred, to reflect the uncertainty of its predictions. Depending on the specific application, our domain voter could be any type of classifier: from a simple naive Bayes for structured data to a deep neural network for image classification.

Note that, conceptually, DAV is in line with crowdsourcing, which was introduced as a cost-saving alternative to expert supervision. DAV considers an extra weighted vote which is obtained *for free*. While the aggregation performance is preserved, DAV requires fewer annotators, thus reducing the cost.

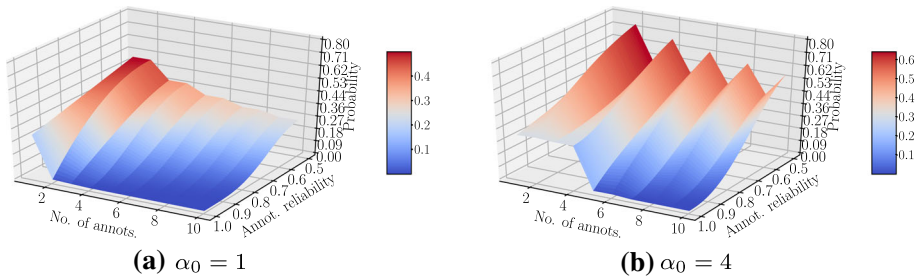
#### 4.1 Intuition on the behaviour of DAV

For the sake of a better understanding of the expected performance of DAV, some insights into its behaviour under different conditions are given hereafter. Here we put the focus on two types of scenarios: (i) Scenarios in which the domain voter may switch the choices made by MV and (ii) scenarios in which DAV is expected to obtain better results than MV, in terms of accuracy.

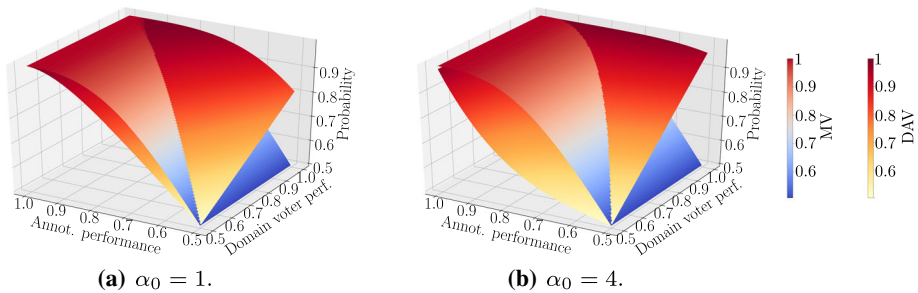
For the sake of simplicity, let us consider a binary class ( $r = 2$ ) and a deterministic domain voter ( $q(c|x) = 1$  for a label  $c$ , and 0 for the rest of labels). Let us define annotator reliability as the probability,  $rel$ , that each annotator selects the correct label. In this binary class context, the most reliable annotator ( $rel = 1$ ) always makes the correct choice; meanwhile, the least valuable contribution comes from those that randomly guess a label ( $rel = 0.5$ ). As we have assumed that annotators provide on average the real class label, we have considered reliability values  $rel \geq 0.5$ .<sup>1</sup> The following results are based on the binomial distribution. Briefly, the probability that  $k$  annotators out of the total number  $m$  (all having reliability  $rel$ ) select the correct class label is  $\binom{m}{k} rel^k (1 - rel)^{m-k}$ .

Firstly, it could be useful to have some insight into when the domain voter can shift the labels provided by MV. In Fig. 2, the probability that the output of the domain voter differs from the one given by MV is depicted. The probabilities estimated for different numbers of annotators (from 1 to 14), reliability values (from 0.5 to 1) and  $\alpha_0$  values (1 in Fig. 2a and 4 in Fig. 2b) are shown, using a domain voter with 0.7 of accuracy. According to Fig. 2, the probability that the domain voter changes the choices made by MV increases (i) as the number of annotators decreases and (ii) as their reliability decreases. On the one hand, as the reliability of the annotators decreases, a lower proportion of them will vote for the same label (higher balance is expected). Thus, there is a higher probability that DAV tips the balance towards the other option. On the other hand, the expected difference between the number of votes gathered by both classes decreases as fewer annotators take part and, again, DAV has higher chances of giving an output different from that of MV. The aforementioned self-regulated behaviour of DAV can be observed: fixed  $\alpha_0$ , the probability of shifting the decision of MV increases as the number of annotators decreases. Finally, note that, when the number

<sup>1</sup> Remember that DAV does not model neither annotator reliability, nor any other characteristic. This reliability concept is an experimental design parameter.



**Fig. 2** Graphical description of the probability that the domain voter changes the choice made by MV, as the number of annotators increases (from 1 to 14) and the reliability of annotators increases (from 0.5 to 1). The value of the parameter  $\alpha_0$  is different for each subfigure and the performance of the domain voter is set to 0.7



**Fig. 3** Graphical description of the expected accuracy values of DAV and MV, as the reliability of annotators increases (from 0.5 to 1) and the performance of the domain voter is equal to that of MV (ranging from 0.5 to 1). The value of the parameter  $\alpha_0$  is different for each subfigure, and the number of annotators is set to 5

of annotators is even, ties may occur when applying MV. In those cases the domain vote would break the tie. This difference explains the stepped behaviour observed in Fig. 2: the contribution of DAV is unquestionably more promising.

These results suggest that the reliability of the annotators has a greater influence than the number of annotators on the probability that DAV changes the answer of MV. The effective difference between both factors rises with large  $\alpha_0$  values (Fig. 2b vs. a). Reliable annotators ( $rel \rightarrow 1$ ) tend to concur voting for the correct label and, intuitively, shifting the choice made by MV is harder. Conversely, almost random annotators ( $rel \rightarrow 0.5$ ) tend to provide both labels at the same rate, and shifting the choice made by MV is more probable.

If the reliability of the annotators and the performance of the domain voter are known, the expected accuracy values of DAV and MV can be computed. That information would be useful to make decisions before applying DAV. In Fig. 3, we compare both methods as the reliability of the annotators and the performance of the domain voter ranges from 0.5 to 1 and the parameter  $\alpha_0$  takes the values 1 and 4 (the number of annotators is set to 5). DAV is expected to outperform MV when annotators are unreliable and the performance of the domain voter is high. The difference increases as the reliability of the annotators decreases and the performance of the domain voter increases. MV outperforms DAV when annotators show intermediate reliability and the domain voter performs poorly. Regarding  $\alpha_0$ , DAV outperforms MV more often in experiments where the weight of the domain voter is lower ( $\alpha_0 = 1$ ). However, the performance differences between DAV and MV are more prominent when the domain voter performs better and it is given a higher weight ( $\alpha_0 = 4$ ). Note that *not all* the scenarios observed in Fig. 3 are necessarily realistic. It is reasonable to expect that

**Table 1** Selected supervised datasets from UCI repository [19]. The columns display, in the following order: Name of the dataset, number of instances ( $n$ ), dimension of the explanatory variable ( $d$ ) and number of classes ( $r$ )

Name	$n$	$d$	$r$	Name	$n$	$d$	$r$
<i>Arrhythmia</i>	452	279	13	<i>Segment</i>	2310	19	7
<i>Dermatology</i>	366	34	6	<i>Vehicle</i>	846	18	4
<i>Glass</i>	214	9	6	<i>Vowel</i>	990	10	11
<i>Satimage</i>	6435	36	6				

the domain voter performs better than a single annotator, as it might simply be built taking into account the labels provided by all annotators. A domain voter with performance much lower than annotator reliability might be unusual in practice.

In the next section, we present a more realistic and extensive comparison between DAV and MV under varying experimental conditions.

## 5 Empirical study

The presented label aggregation scheme, DAV, is proposed as an enhancement of MV that incorporates extra information, from the descriptive variable and all the instances. We simulate a large spectrum of scenarios and aim to identify those in which DAV outperforms MV. Scenarios where instances might be labelled by few or no annotators, where these have varying reliability values, are considered. Each experiment is run 100 times, and the mean values of the accuracy are reported.

Our hypotheses are: (H1) There exists an  $\alpha_0 > 0$  for every dataset that makes DAV better (or at least not worse) than MV in terms of aggregation accuracy (Eq. 1), (H2) the advantage of DAV regarding MV tends to increase as the number of labels collected for each instance decreases, and (H3) the advantage of DAV regarding MV increases as the reliability of the annotators decreases. We validate these hypotheses with (i) standard supervised data and synthetic annotators and (ii) real crowdsourced data in the following subsections.

### 5.1 Experiments with artificial annotations on standard supervised datasets

Firstly, we consider fully supervised datasets and synthetically transform them into crowdsourced labelled datasets employing simulated annotations. This allows us to control the reliability of the annotators and thus to validate Hypothesis H3.

We consider datasets with different numbers of instances, class labels, and dimensions of the explanatory variable, to cover a variety of experimental scenarios. In that way, the strengths and weaknesses of DAV concerning the baseline MV can be observed accounting for a wide range of characteristics. The datasets, collected from the UCI repository [19], and their main characteristics are summarised in Table 1.

#### 5.1.1 Artificial labels generation

To generate meaningful synthetic labels for each dataset, we take into account the class-confusion matrix of a random forest (RF) classifier [20]. a reliability parameter ( $rel$ ) sets the probability that an annotator labels an instance correctly, and is used to simulate the mistakes of the annotators.

To generate the multiple noisy labels, the following procedure is carried out. Given a supervised dataset, we use stratified 10-fold cross-validation [21] to estimate the mean class confusion matrix  $M$  of a RF model learned from it. The rows of  $M$  are normalised so that they all add up to 1. Then, a matrix  $R$  is constructed as follows. For  $c \in \{1, \dots, r\}$ :

- $R_{c,c} = rel$
- For  $c' \neq c$ :
  - $R_{c,c'} = \frac{M_{c,c'}(1-rel)}{\sum_{c'' \neq c} M_{c,c''}}$ , if  $\exists c'' \neq c: M_{c,c''} > 0$ .
  - $R_{c,c'} = \frac{1-rel}{r-1}$ , otherwise.

In this way, the element  $R_{c,c'}$  is the probability that an annotator assigns the label  $c'$  to an instance of real class  $c$ . The annotator model is consistent with the specified annotator reliability, as  $rel = R_{c,c}$ , and with the confusion between classes estimated in matrix  $M$ .

An annotation for an instance of class  $c_x$  can be simulated by sampling the distribution  $R_{c_x} = (R_{c_x,1}, \dots, R_{c_x,r})$ . To obtain several artificial annotations, the distribution  $R_{c_x}$  is independently sampled. As our goal is not to model the annotators, we do not consider differences between them: all of them are simulated through the same matrix,  $R$ . Also, for the sake of simplicity, the same number of labels,  $l$ , is sampled for each instance. Given an instance  $x$  with real class  $c_x$ , the distribution  $R_{c_x}$  is sampled  $l$  times, and the obtained labels form the collection  $S_x$ .

### 5.1.2 Label sets of different sizes

Crowdsourced datasets usually have instances with different numbers of labels (some even with very few labels or none), a scenario strongly related to hypothesis H2. To consider this in our experiments, the label sets of the instances might be transformed in three different ways:

**Config. A** The datasets are used with all the sampled labels.

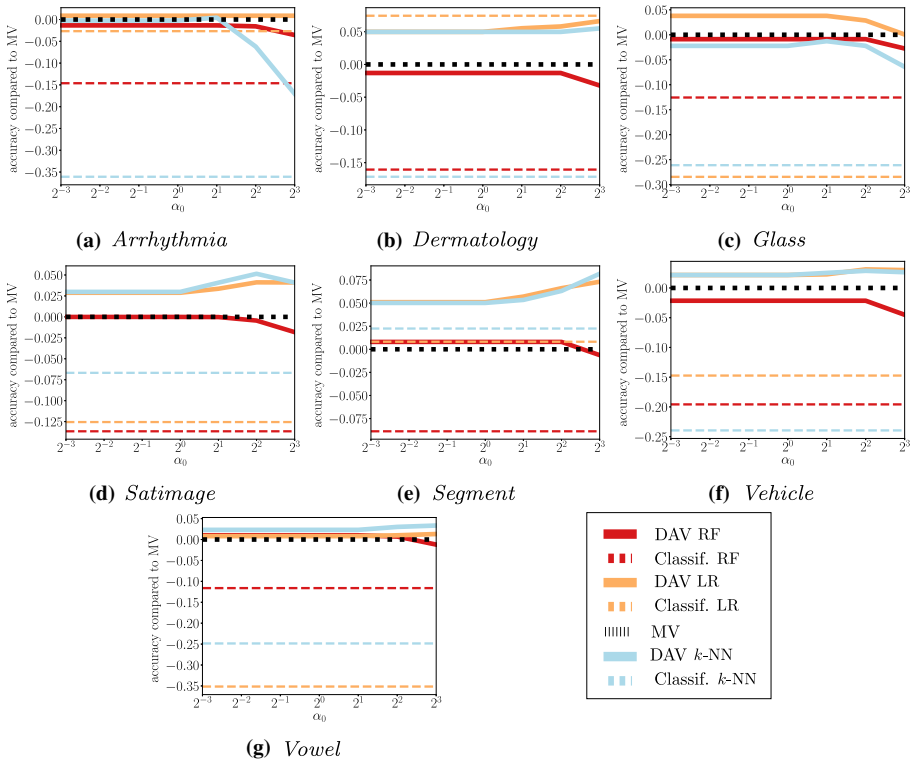
**Config. B** All labels assigned to a specific subset of the instances are discarded.

**Config. C** Labels are randomly discarded (uniformly or not).

For configuration B, the proportion of instances whose assigned label sets are emptied is controlled by a parameter  $p_d$ . In practice, labels are discarded as follows: An instance is randomly selected with probability  $p_d$ . Next, all the labels of the selected instances are discarded. The expected number of instances whose labels are discarded is  $|D| \cdot p_d$ . By assigning different values to  $p_d$ , the robustness of the methods in front of datasets with unlabelled examples can be observed.

For configuration C, a concentration parameter (*con*) controls the variance of the number of discarded labels for different instances. The proportion of labels to eliminate for each particular instance is determined by a Beta distribution. In practice, labels are removed as follows: Given an instance  $x$ , each label in the collection  $S_x$  is discarded with probability  $\beta_x \sim B(con, con)$ . Since the two parameters of the Beta distribution are equal, the expected average number of discarded labels is  $\frac{|S_x|}{2}$ . When  $con = 1$ , all the numbers of labels to discard in the range  $\{0, \dots, |S_x|\}$  have the same probability. As  $con \rightarrow 0$ , the number of eliminated labels tends to be extreme (closer to either 0 or  $|S_x|$ ), i.e., the variance tends to its maximum. As  $con \rightarrow \infty$ , the number of discarded labels gets closer to the mean  $\frac{|S_x|}{2}$ , i.e., the variance tends to 0. By varying the value of the parameter *con*, scenarios where there is a fixed budget but the annotations are distributed throughout the instances in different ways can be observed.





**Fig. 4** Graphical description of the accuracy obtained by DAV with different classifiers and the classifiers themselves compared to the accuracy of MV, as the weight of the domain voter ( $\alpha_0$ ) increases,  $\alpha_0 = 2^e$  with  $e \in \{-3, \dots, 3\}$ . Results obtained with artificial annotations on supervised datasets are displayed, using all labels (configuration A). The values of the rest of the parameters are fixed:  $l = 6$  (number of labels per instance) and  $rel = 0.7$  (reliability of the annotators)

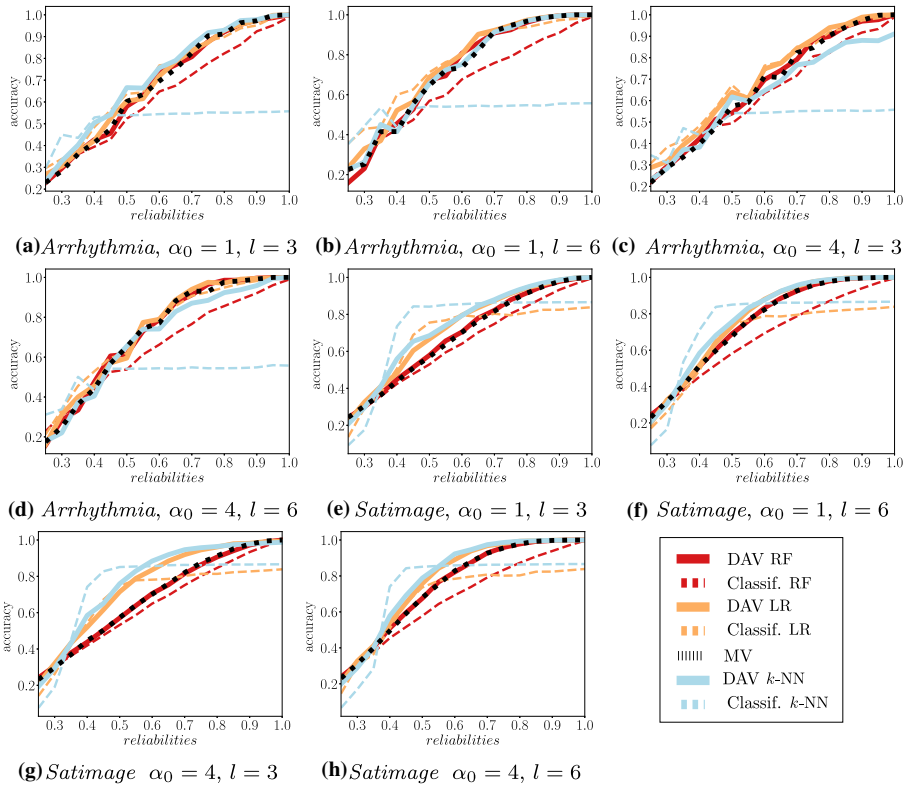
### 5.1.3 Implementation of DAV

*Domain voter building* Three models have been selected as domain voters:  $k$ -nearest neighbours ( $k$ -NN), logistic regression (LR) and random forest (RF). The domain voter is trained using all the annotated instances. In particular, the instances have probabilistic labels corresponding to their voting estimate (see Eq. 3).

*Operating DAV* Given an instance  $x$ , the domain voter is used to get a distribution over the classes and the voting estimate (Eq. 3) is computed for all classes. Both are combined computing the DAV estimate as in Eq. 5, and the argument of the maximum is taken as the result (Eq. 4).

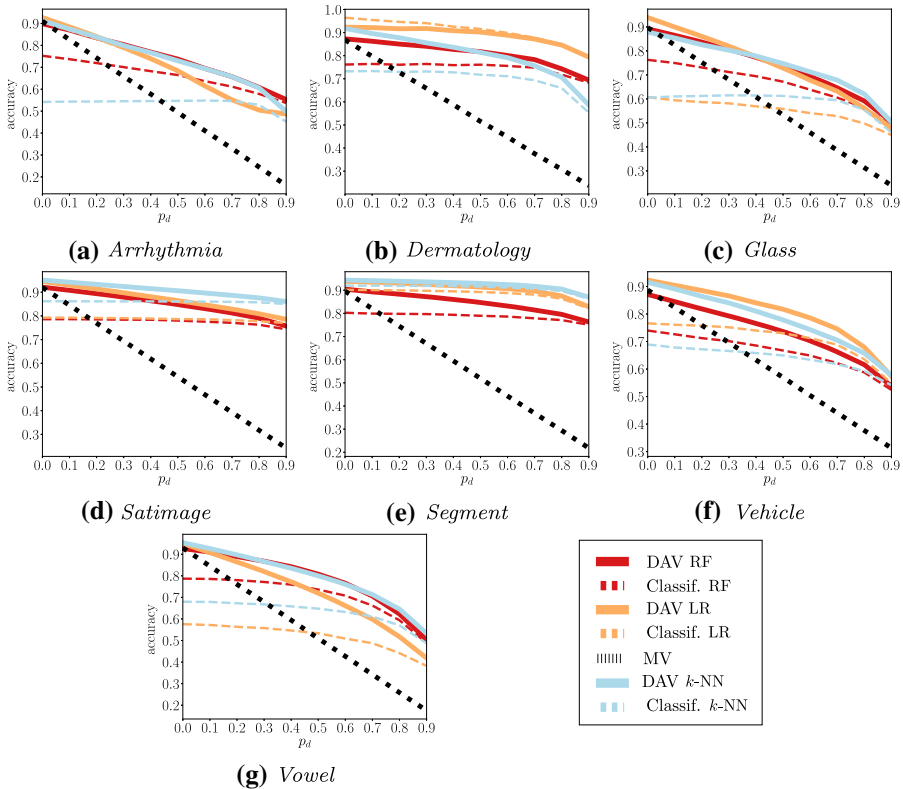
### 5.1.4 Experimental results with supervised datasets

The results obtained with supervised datasets (Table 1) and under different experimental conditions are discussed below. Inspired by real scenarios (see Sect. 5.2), we fix  $l = 6$  simulated labels for each instance from the supervised datasets. Each experiment is run 100 times, and the mean values of the accuracies are obtained.



**Fig. 5** Graphical description of the accuracy obtained by MV, DAV with different classifiers and the classifiers themselves, as the value of the parameter  $rel$  (reliability of the annotators) increases,  $rel \in \{0.25, 0.3, \dots, 1\}$ . Results obtained with artificial annotations on the supervised datasets *arrhythmia* and *satimage* are displayed, using the complete labellings (configuration A). Specific configurations (dataset and values of  $\alpha_0$  and  $l$ ) are used in each subfigure, as detailed in their captions

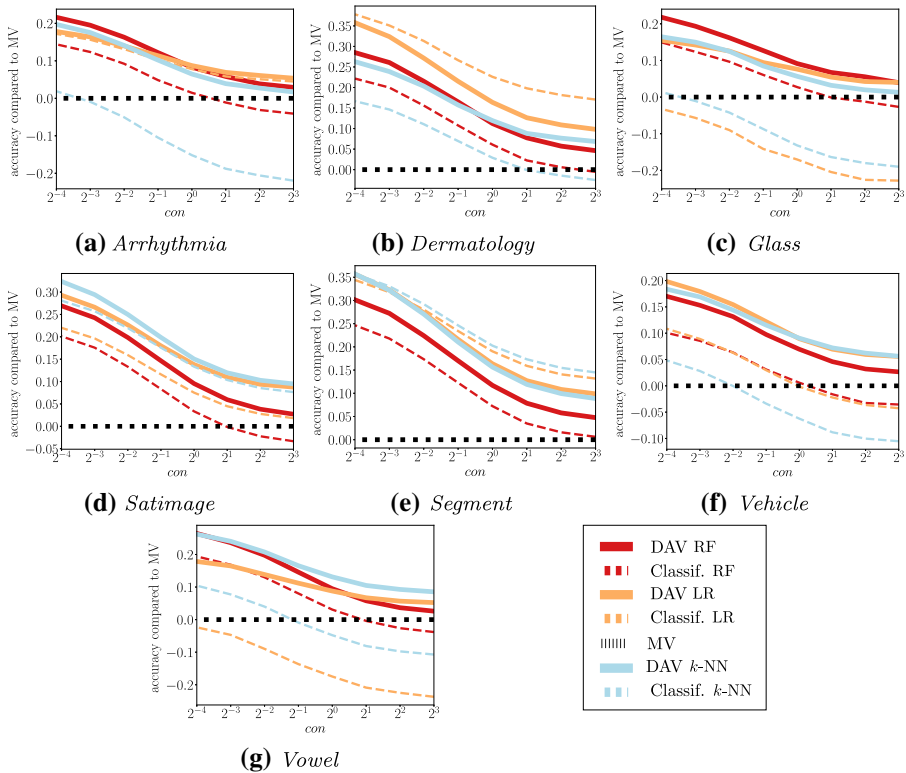
In Fig. 4, the evolution of the mean accuracy with respect to the weight of the domain voter ( $\alpha_0$ ) can be observed. The value of  $\alpha_0$  ranges from  $2^{-3}$  (when DAV closest resembles MV) to  $2^3$  in a logarithmic scale, without discarding any label (configuration A). The reliability parameter  $rel$  is set to 0.7. DAV achieves a better (or at least equal) performance than MV in all the datasets, as there always exists a value of  $\alpha_0$  and a classifier for each dataset that allows DAV to outperform MV. Summing up through the different combinations of datasets and classifiers, DAV outperforms MV in 19 out of the 21 experiments. Note that DAV obtains a higher accuracy than the domain voter in 20 out of the 21 experiments. When a classifier obtains a lower accuracy than MV, in most cases, the accuracy of DAV gets closer to that of MV as the weight of the domain voter increases. However, there are cases where the accuracy of DAV increases as the weight of the domain voter increases, such as the datasets *dermatology* (Fig. 4b), *satimage* (Fig. 4d) and *segment* (Fig. 4e). Thus, by using a selection criterion for the value of  $\alpha_0$  (as discussed in Sect. 6.2), a setup that leads to equal or better performance than that of MV can be achieved. These results are in line with our Hypothesis H1. Note that the domain voters have a poorer performance than MV in almost all the scenarios observed in Fig. 4. Nevertheless, DAV is still able to outperform MV in most cases: The extra information



**Fig. 6** Graphical description of the accuracy obtained by MV, DAV with different classifiers and the classifiers themselves, as the value of the parameter  $p_d$  (configuration B) increases,  $p_d \in \{0, 0.1, \dots, 0.9\}$ . Results obtained with artificial annotations on supervised datasets are displayed. The values of the rest of the parameters are fixed:  $\alpha_0 = 1$ ,  $l = 6$  (maximum number of labels per instance) and  $rel = 0.7$  (reliability of the annotators)

incorporated by DAV seems to complement the plain aggregation of labels. Moreover, DAV used with the  $k$ -NN model leads to the best results in almost all experiments, even though that classifier has an overall poorer performance than the other ones.

Figure 5 shows the evolution of the mean accuracy of the methods with respect to the reliability of the annotators,  $rel \in \{0.25, 0.3, \dots, 1\}$ , and considering different values for parameters  $\alpha_0$  and  $l$ . We concentrate in two datasets: *arrhythmia* and *satimage*, as they show similar trends to the results on other datasets). As the reliability of the annotators increases, so do the accuracy values of DAV and MV. The accuracies of DAV and MV are very similar for extreme values of  $rel$  in most scenarios. DAV reaches a better performance than the domain voters for most levels of annotator reliability, except for the lowest values in *arrhythmia* dataset (Figs. 5a to d) and for medium values in *satimage* dataset (Figs. 5e to f). With *arrhythmia*, the reliability of the annotators does not have a visible influence in the differences between the accuracy values of the studied methods, as opposed to *satimage*. Moreover, in the cases where the reliability affects the difference between the accuracy values of DAV and MV, this increases quickly with low reliability values, and then reduces smoothly. This behaviour is related to our hypothesis H3, as there is a greater difference for non-extreme low reliability ( $rel$ ) annotators. Similarly to the previous one, Fig. 5 shows that the mean accuracy of the domain voter is lower than that of MV in almost all the experiments.



**Fig. 7** Graphical description of the accuracy obtained by DAV with different classifiers and the classifiers themselves compared to the accuracy of MV, as the value of the parameter  $con$  (configuration C) increases,  $con = 2^e$  with  $e \in \{-4, \dots, 3\}$ . Results obtained with artificial labels on supervised datasets are displayed. The values of the rest of the parameters are fixed:  $\alpha_0 = 1$ ,  $l = 6$  (max. no. of labels per instance) and  $rel = 0.7$  (reliability of the annotators)

In Fig. 6, configuration B (Sect. 5.1.2), where the label set of each instance is emptied with probability  $p_d$ , is studied. The values of  $p_d$  range from 0 to 0.9 and the value of  $\alpha_0$  is set to 1, i.e., the domain voter has the same weight as any other annotator. As the proportion of non-annotated instances ( $p_d$ ) grows linearly, the performance difference between DAV and MV grows linearly as well, until the proportion of unlabelled instances reaches 0.5 – 0.7. Then, in most cases, that difference slightly decreases, with a few exceptions (Figs. 6a and 6d). That is, DAV does not seem to be affected by the lack of labels as much as MV does, which supports our hypothesis H2. Note that, as the proportion of unlabelled instances ( $p_d$ ) grows, the accuracy of each classifier gets closer to the accuracy of DAV obtained with that classifier. This behaviour is related to the fact that DAV provides the same label as the domain voter for unlabelled instances.

Results under experimental configuration C are displayed in Fig. 7. The evolution of the accuracy with respect to the concentration of labels ( $con$ ) (values  $2^e$  where  $e \in \{-4, \dots, 3\}$ ) can be observed. The rest of the parameters are fixed:  $\alpha_0 = 1$  and  $rel = 0.7$ . Let us recall the effect of parameter  $con$  in the distribution of labels: When the parameter  $con$  has low values, half of the instances tend to lose all their labels; when  $con$  is high, all the instances tend to lose half of their labels. In this way, the effect of the lack of labels is observed in the

**Table 2** Selected crowdsourced datasets. The columns display, in the following order: Name of the dataset, number of instances ( $n$ ), dimension of the explanatory variable ( $d$ ), number of classes ( $r$ ), number of annotators and mean number of labels per instance. The dataset *music genre* is from [22], the datasets of *Quality assessment (QA)* are from the UCI repository and *sentiment polarity* was introduced by [23]

Name	$n$	$d$	$r$	# annot.	mean # labels
<i>music genre</i>	700	124	10	44	4.21
<i>QA: Green</i>	98	62	2	6	6
<i>QA: Hinselmann</i>	97	62	2	6	6
<i>QA: Schiller</i>	92	62	2	6	6
<i>Sentiment polarity</i>	4999	1200	2	203	2.55

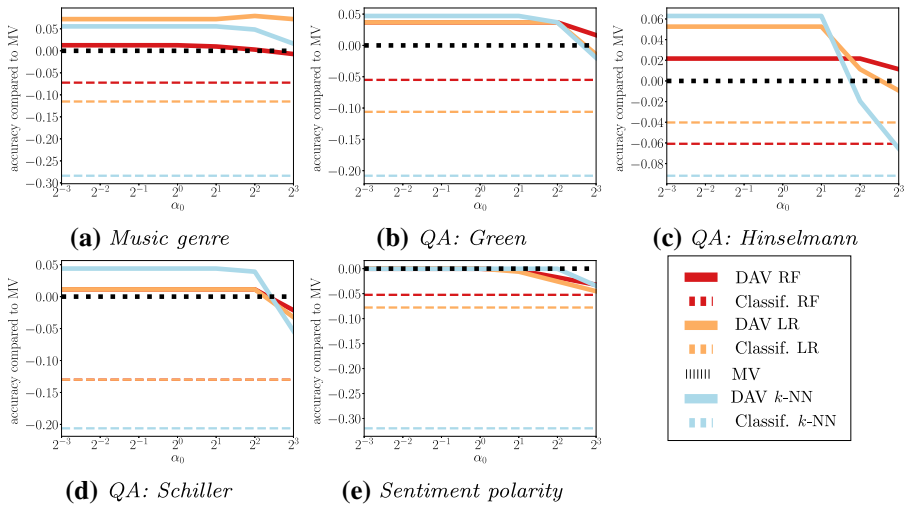
whole spectrum between the two aforementioned scenarios. The average difference between the performances of DAV and MV observed in Fig. 7 is greater than the one observed in Fig. 4. This fact matches Hypothesis H2 since fewer labels are collected in average in configuration C (Fig. 7) than in configuration A (Fig. 4). Moreover, the difference between the accuracy values of the two methods is larger when a subset of instances is unlabelled (low values of  $con$ ) than when all the instances provided have fewer labels (high values of  $con$ ). Indeed, this is related to the self-regulatory behaviour of DAV: Given a weight for the domain voter ( $\alpha_0$ ), the domain vote gains importance over the votes of the annotators as the number of available labels decreases. It is again noteworthy that, even when a classifier reaches a poorer performance than MV, DAV outperforms MV when using that classifier as domain voter.

## 5.2 Experimental results with real-world crowdsourced datasets

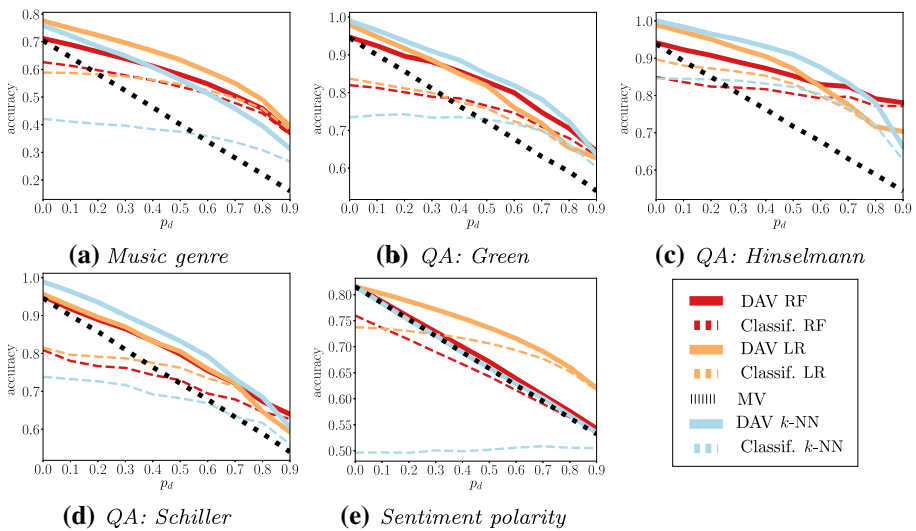
In this second set of experiments, real crowdsourced datasets are used to test our hypotheses. Datasets with different numbers of annotators and mean numbers of labels per instance have been considered, as summarised in Table 2. A similar experimental setting as in the previous subsection is followed. It only differs in the fact that, in this new set of experiments, real crowd annotations are available and their simulation is not needed.

Figure 8 shows the evolution of the accuracy with respect to the weight of the domain voter ( $\alpha_0$ ), which ranges from  $2^{-3}$  (when DAV closest matches the behaviour of MV) to  $2^3$  in a logarithmic scale, without discarding any label (configuration A). According to Fig. 8, H1 seems to be supported as DAV outperforms or at least equals the performance of MV for  $\alpha_0 \leq 2$  on all the considered datasets and classifiers. Moreover, the average difference between the accuracy values of the two methods seems to be higher. Again, the weight of the domain voter ( $\alpha_0$ ) increases, it gains more importance over the crowdsourced labels, and the accuracy of DAV tends to that of the classifier. If the accuracy of the classifier is lower than that of MV, it may affect DAV resulting in a worse performance than MV. As aforementioned, the results suggest that an equal or better accuracy than that of MV can be achieved with DAV for certain values of the parameter  $\alpha_0$ .

In Fig. 9, the results for experimental configuration B (Sect. 5.1.2) are displayed, where all the annotations of each instance are discarded with probability  $p_d \in \{0, 0.1, \dots, 0.9\}$ . Recall that, when  $p_d = 0$ , all the labels of each dataset are included. The rest of the parameters are fixed:  $\alpha_0 = 1$  and  $rel = 0.7$ . In that figure, similar patterns to those observed in the real crowd datasets can be seen (Fig. 6 in Sect. 5.1.4). The increase in the difference between



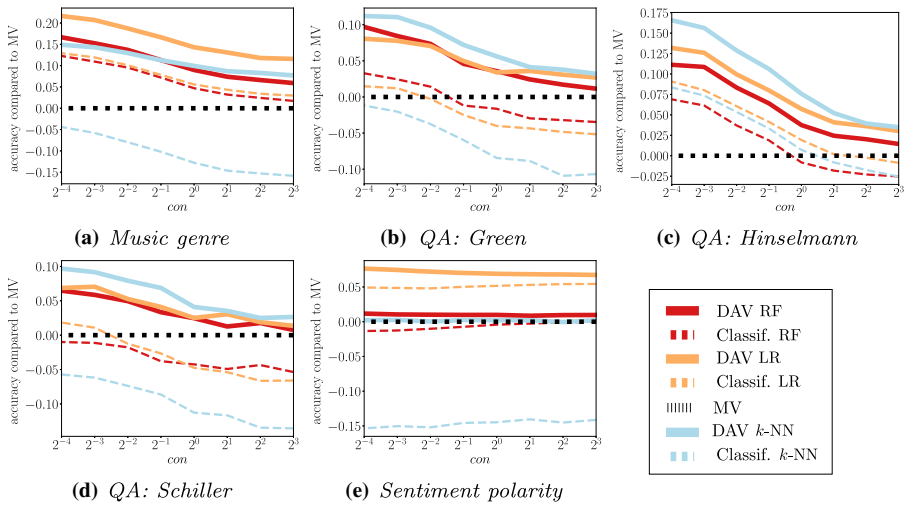
**Fig. 8** Graphical description of the accuracy obtained by DAV with different classifiers and the classifiers themselves compared to the accuracy of MV, as the weight of the domain voter ( $\alpha_0 = 2^e$  with  $e \in \{-3, \dots, 3\}$ ). Results obtained with real crowdsourced datasets are displayed



**Fig. 9** Graphical description of the accuracy obtained by MV, DAV with different classifiers and the classifiers themselves, as the value of the parameter  $p_d$  (configuration B) increases,  $p_d \in \{0, 0.1, \dots, 0.9\}$ . Results obtained with annotations of real crowdsourced datasets are displayed. The value of  $\alpha_0$  is set to 1

the accuracy values of DAV and MV is almost linear with respect to the evolution of the parameter  $p_d$ , with a small drop for  $p_d \geq 0.7$ , in almost every scenario.

Configuration C is considered in Fig. 10. Labels are discarded depending on a Beta distribution  $B(con, con)$  as explained in Sect. 5.1.2 and the results are displayed for different values of the concentration of labels ( $con$ ) (values  $2^e$  where  $e \in \{-4, \dots, 3\}$ ). The value of  $\alpha_0$  is fixed to 1. The results match those observed in the experimental results obtained



**Fig. 10** Graphical description of the accuracy obtained by DAV with different classifiers and the classifiers themselves compared to the accuracy of MV, as the value of the parameter  $con$  (configuration C) increases,  $con = 2^e$  with  $e \in \{-4, \dots, 3\}$ . Values of  $con$  are  $2^e$ , starting with  $e = 3$  and decreasing to  $e = -4$ . Results obtained with annotations of real crowdsourced datasets (Table 2) and parameter  $\alpha_0 = 1$  are displayed

with artificial labels, although the differences between the accuracy values of DAV and MV are more limited in this case. A larger difference between the accuracy values of DAV and MV can be observed when there is a lack of labels than when all instances are provided  $l = 6$  labels (Fig. 8), which would support our hypothesis H2. Furthermore, that difference increases when all labels are concentrated in a part of the dataset (low values of parameter  $con$ ), which is a similar scenario to configuration B (Fig. 9).

Overall, the results obtained in this set of experiments are in line with those with synthetic data. Once again it is noteworthy that DAV outperforms MV even when its underlying classifier does not show better results than MV. Similarly, DAV obtains higher accuracy than the domain voter in all the studied scenarios.

## 6 Discussion

Our DAV method can be a promising tool for tackling label aggregation in learning from crowd environments. Evidence collected through two sets of experiments seem to support our three working hypothesis:

- H1 Results in Figs. 4 and 8 show that, for each dataset and classifier, there is at least one value of  $\alpha_0 > 0$  such that DAV outperforms or equals the accuracy of MV.
- H2 Results in Figs. 6, 7, 9 and 10 show that the advantage of DAV over MV increases when there are fewer labels available.
- H3 Results in Fig. 5 show that there is a greater advantage of DAV over MV for (non-extreme) low reliability values.

When applying DAV, several decisions such as the method to obtain the domain votes or how to select the value for  $\alpha_0$  must be made. The ideal way of making those decisions would be by selecting the values that lead to the best performance of DAV. Unfortunately,

this involves the estimation of the performance in the context of crowdsourced labelled data, which is an unsolved problem with a short related literature (e.g., [24]). A few guidelines are offered below on the way of obtaining the domain votes and the selection of a value for  $\alpha_0$ , including other issues.

Some of these guidelines might require an *uncertainty* measure for quantifying how sure we are about the consensus label obtained for a given instance. One could use the entropy of the DAV estimate over the class labels of each instance, taking into account the number of collected labels. But this is not enough, as even although an instance with a single label would have entropy equal to 0, this label might be mistaken since annotators are not expert. One could, instead, perform Bayesian estimation using Dirichlet priors with all hyperparameters equal to 1. Another option could be the Label and Model Uncertainty (LMU) proposed by [2]. In this framework, considering a binary class, the Label Uncertainty (LU) is computed as the tail probability below the labelling decision threshold, assuming that the posterior probability over the true label follows a Beta distribution whose parameters depend on the numbers of both positive and negative votes. The Model Uncertainty (MU) is a score that uses classifiers trained on the available data, and the LMU is computed as the geometric mean of the LU and the MU.

## 6.1 Construction of the domain votes

A key contribution to DAV comes from the domain voter. In the experiments presented in this work, the domain voter is a classifier. We suggest to use the best available classifier in the state of the art for the domain of the problem at hand. Currently, all the instances are considered, with the same weight, to obtain the domain votes. However, one could use an uncertainty measure as aforementioned to identify certainly labelled examples. Instances with highly certain labelling could be given larger weight when building the domain voter, and the other way around. In the particular case that a subset of the instances is fully supervised (completely reliable), the domain voter could be obtained from this subset only. This is evident, for example, in the medical domain where intrusive practices such as punctures or biopsies are limited to a subset of patients. Techniques of semi-supervised learning [25, 26] could also be used to learn from a larger subset including the supervised examples. Finally, if the use of DAV reduces the uncertainty surrounding a specific subset of the instance space, the domain votes could be re-computed including that subset. This reveals a possible iterative application of DAV: The domain votes could be re-computed using the labels obtained through DAV, then perform DAV with the new domain votes, and so on.

## 6.2 Criteria for the selection of $\alpha_0$

One of the main findings from our experiments is that the value of  $\alpha_0$  is determinant and it has to be adjusted for the successful performance of DAV.

There is no straightforward way to choose the optimal value for  $\alpha_0$ . As aforementioned, selecting  $\alpha_0$  using cross-validation is unfeasible. Taking that into account, a few guidelines on the selection of the value of  $\alpha_0$  are as follows:

- Since  $\alpha_0$  controls the weight of the domain votes on DAV, one could pay attention to the performance of the domain voter. When the performance of the domain votes increases, the value of  $\alpha_0$  should be higher, and the other way around. If the performance of the domain votes can be estimated, it can help us make this decision.



- As the mean reliability of the annotators increases, the relative performance of the domain voter is reduced and a lower value for  $\alpha_0$  could be chosen. In that case, the self-regulatory behaviour of DAV would cause a shift in the choices of MV only in instances with few labels or tied voting. Annotator models [11, 12, 27] could be used to estimate those reliability values.

As many of these concepts (good/bad performance, low/high uncertainty) are subjective, the final user has to choose among the considered scenarios and recommendations based on their own judgement.

### 6.3 DAV in dynamic environments

Note that the scenario considered in this work is static: All of the instances and labels are available from the beginning. All of them are then used to obtain the domain voter, which is used to enhance the label aggregation process.

However, in many real-world applications, the environment is dynamic, i.e., new instances and/or labels may be gathered after the domain votes were computed. Different such examples include online learning, where instances come sequentially and not in a single batch from the beginning, and active learning [28], where new labels can be requested for specific instances. In these dynamic scenarios, the ideal strategy would be to re-compute the domain votes for every new piece of information (instance and/or label), as it is always beneficial for DAV. However, the methods for obtaining the domain votes could be excessively costly regarding the available resources. Thus, to adapt DAV to dynamic environments, one should consider whether the domain voter needs to be re-computed or not at every single step. To make that decision, one could use one of the aforementioned uncertainty measures in order to quantify the information gathered since the last update. For example, a new instance with low uncertainty or a new label that reduces the uncertainty of an instance would bring more information than an instance with higher uncertainty or a label which increases the uncertainty of an instance. When the amount of information brought by the new instances (or labels) is sufficiently high, the domain votes should be computed again including the new data in the dataset. The parameter  $\alpha_0$  could be tuned accordingly as well.

## 7 Conclusions and future work

In this work, domain-aware voting (DAV), a novel method for crowdsourced label aggregation, is presented. As opposed to majority voting, it uses information from the entire dataset and the descriptive variable by means of an extra weighted vote.

Empirical evidence, which was obtained through a vast experimental setting, supports our three hypotheses: (i) there exists a weight for the domain vote for every dataset that makes DAV competitive regarding MV, (ii) DAV outperforms MV more largely as the number of annotations per instance decreases, and (iii) the difference becomes bigger as the reliability of the annotators decreases. Thus, DAV arises as a useful alternative to MV, especially for scenarios where labels are scarce. DAV also exhibits an interesting self-regulated behaviour: The importance of the domain vote increases as the number of annotations decreases, and vice versa. As a consequence of the enhanced efficiency of DAV (its results are better with fewer annotations), the budget for crowdsourced labelling might be reduced.

We also provide practical guidelines on how to set DAV parameters. In the future, it would be interesting to work on a robust method to select a value for the relevant  $\alpha_0$  parameter in

a more informed way. Another next step would be to consider other domain voters, such as using prior probabilities or density estimation based on previously observed data. Moreover, DAV could be easily adapted for dynamic environments or to work as an intermediate step of more sophisticated techniques. It would be particularly interesting to develop techniques that involve modelling the annotators. Having an annotator model can serve to weigh their contribution or to detect and correct adversarial or colluding behaviours.

**Acknowledgements** This work was partially supported by Spanish Ministry of Science, Innovation and Universities through BCAM Severo Ochoa excellence accreditation SEV-2017-0718; by Basque Government through BERC 2022-2025 and ELKARTEK programs. During a large part of this work, IBM held a grant no. BES-2016-078095. JHG is a Serra Hünter Fellow. The authors would also like to thank Dr. Jesús Cerquides (IIIA-CSIC) for his helpful comments.

**Funding** Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

1. Snow R, O'Connor B, Jurafsky D, Ng AY (2008) Cheap and fast - but is it good? evaluating non-expert annotations for natural language tasks In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp 254–263
2. Sheng VS, Provost FJ, Ipeirotis PG (2008) Get another label? improving data quality and data mining using multiple, noisy labelers In: Proceedings of the Special Interest Group on Knowledge Discovery and Data Mining, pp 614–622
3. Fix E, Hodges JL Jr (1951) Discriminatory analysis-nonparametric discrimination: consistency properties. California Univ Berkeley, Tech. rep
4. Abououf M, Otrok H, Mizouni R, Singh S, Damiani E (2020) How artificial intelligence and mobile crowd sourcing are inextricably intertwined. *IEEE Netw* 35(3):252–258
5. Sheng VS, Zhang J (2019) Machine learning with crowdsourcing: a brief summary of the past research and future directions. *Proc AAAI Conf Artif Intel* 33:9837–9843
6. Bernstein MS, Little G, Miller RC, Hartmann B, Ackerman MS, Karger DR, Crowell D, Panovich K (2015) Soylent: a word processor with a crowd inside. *Commun Assoc Comput Mach* 58(8):85–94
7. Corney J, Lynn A, Torres C, Di Maio P, Regli W, Forbes G, Tobin L (2010) Towards crowdsourcing translation tasks in library cataloguing, a pilot study In: IEEE International Conference on Digital Ecosystems and Technologies, IEEE, pp 572–577
8. Wazny K (2018) Applications of crowdsourcing in health: an overview *J Global Health* 8(1)
9. Rodrigo GE, Aledo JA, Gámez JA (2019) Machine learning from crowds: a systematic review of its applications. *Wiley Interdiscip Rev Data Mining Knowl Discov* 9(2):e1288
10. Sheng VS, Zhang J, Gu B, Wu X (2017) Majority voting and pairing with multiple noisy labeling. *IEEE Trans Knowl Data Eng* 31(7):1355–1368
11. Karger DR, Oh S, Shah D (2011) Iterative learning for reliable crowdsourcing systems *Neural Information Process Syst* pp 1953–1961
12. Dawid AP, Skene AM (1979) Maximum likelihood estimation of observer error-rates using the EM algorithm. *J Roy Stat Soc Ser C (Appl Stat)* 28(1):20–28
13. Zhang Y, Chen X, Zhou D, Jordan MI (2016) Spectral methods meet EM: a provably optimal algorithm for crowdsourcing. *J Mach Learn Res* 17(1):3537–3580
14. Rodrigues F, Pereira F (2018) Deep learning from crowds In: Proceedings of the AAAI Conference on Artificial Intelligence, vol 32

15. Tanno R, Saeedi A, Sankaranarayanan S, Alexander DC, Silberman N (2019) Learning from noisy labels by regularized estimation of annotator confusion In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 11244–11253
16. Raykar VC, Yu S, Zhao LH, Valadez GH, Florin C, Bogoni L, Moy L (2010) Learning from crowds. *J Mach Learn Res* 11:1297–1322
17. Yan Yea (2010) Modeling annotator expertise: learning when everybody knows a bit of something In: Proceedings of AISTATS, pp 932–939
18. Zhang J, Sheng VS, Wu J (2019) Crowdsourced label aggregation using bilayer collaborative clustering. *IEEE Transactions Neural Netw Learn Syst* 30(10):3172–3185
19. Frank A, Asuncion A (2010) UCI machine learning repository <http://archive.ics.uci.edu/ml>
20. Breiman L (2001) Random forests. *Mach Learn* 45(1):5–32
21. Rodríguez JD, Pérez A, Lozano JA (2013) A general framework for the statistical analysis of the sources of variance for classification error estimators. *Pattern Recogn* 46(3):855–864
22. Tzanetakis G, Cook P (2002) Musical genre classification of audio signals. *IEEE Transactions Speech Audio Process* 10(5):293–302
23. Pang B, Lee L (2005) Seeing stars: exploiting class relationships for sentiment categorization with respect to rating scales In: Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, pp 115–124
24. Urkullu A, Pérez A, Calvo B (2019) On the evaluation and selection of classifier learning algorithms with crowdsourced data. *Appl Soft Comput* 80:832–844
25. Chapelle O, Scholkopf B, Zien A (2009) Semi-supervised learning. *IEEE Transactions Neural Netw Learn Syst* 20(3):542–542
26. Zhu XJ (2005) Semi-supervised learning literature survey University of Wisconsin-Madison Department of Computer Sciences, Tech Rep
27. Whitehill J, fan Wu T, Bergsma J, Movellan JR, Ruvolo PL (2009) Whose vote should count more: optimal integration of labels from labelers of unknown expertise *Neural Information Process Syst* pp 2035–2043
28. Yan Y, Rosales R, Fung G, Dy JG (2011) Active learning from crowds. *Int Conf Mach Learn* 11:1161–1168

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Iker Beñaran-Muñoz** is currently a PhD student at University of the Basque Country (UPV/EHU). His research interests include learning from crowdsourced data. He finished a Master's Degree in Computational Engineering and Intelligent Systems at the UPV/EHU in September 2016. Contact him at [iker.benaran@gmail.com](mailto:iker.benaran@gmail.com).



**Jerónimo Hernández-González** is currently a Serra Húnter (tenure-eligible) lecturer in the department of Mathematics and Computer Science at University of Barcelona, Spain. His major research interests include weak supervision, learning and inference with probabilistic graphical models, and their applications to biomedical and educational domains. He received his PhD in computer science from the University of the Basque Country, Spain, in 2015. Contact him at [jeronimo.hernandez@ub.edu](mailto:jeronimo.hernandez@ub.edu).



**Aritz Pérez** is currently a postdoctoral researcher at the Basque Center for Applied Mathematics. His main research lines include supervised, unsupervised and weak classification, probabilistic graphical models, and time-series data mining, with applications to industry, energy management and health care. He received his PhD degree in 2010 from the the University of Basque Country. Contact him at [aperez@bcamath.org](mailto:aperez@bcamath.org).