

# UNSUPERVISED ENSEMBLE CLASSIFICATION WITH CORRELATED DECISION AGENTS

Margarita Cabrera-Bean, *Senior Member, IEEE*, Alba Pagès-Zamora, *Senior Member, IEEE*, and Carles Díaz-Vilor

**Abstract**—Decision-making procedures when a set of individual binary labels is processed to produce a unique joint decision can be approached modeling the individual labels as multivariate independent Bernoulli random variables. This probabilistic model allows an unsupervised solution using EM-based algorithms, which basically estimate the distribution model parameters and take a joint decision using a Maximum a Posteriori criterion. These methods usually assume that individual decision agents are conditionally independent, an assumption that might not hold in practical setups. Therefore, in this work we formulate and solve the decision-making problem using an EM-based approach but assuming correlated decision agents. Improved performance is obtained on synthetic and real datasets, compared to classical and state-of-the-art algorithms.

**Index Terms**—Correlated Bernoulli Distribution, Crowdsourcing, Unsupervised Ensemble Learning, Correlated Decision Agents.

## I. INTRODUCTION

WITH the advent of crowdsourcing, ensemble learning has recently regained interest as a useful methodology to rate annotators and to combine the data they provide, accordingly. In particular, unsupervised ensemble classification refers to the problem of designing a meta-learner to classify objects without the help of training data and based only on labels provided by individual decision agents [1]. Applications are found in diverse areas such as medicine and biology, e.g. [2][3] where decision agents are either algorithmic techniques or individuals, respectively; team decision-making strategies [4]; and 5G communication systems [5], where decision agents are sensors.

Based on the seminal paper of Dawid and Skene [6], a vast majority of recent works, e.g. [7],[8],[9], estimate the statistics of the decision agents by different means, and use these estimates either to solve a Maximum Likelihood (ML) detection problem or to run the Expectation-Maximization (EM) algorithm in [6] initialized with these estimates. In [10], a Maximum a Posteriori (MAP) approach is adopted to solve a multi-class ensemble classification problem using moment-based estimates of the confusion matrices of the decision agents. Interestingly, an upper bound of the expected error rate is derived in [11][12] for the EM algorithm and MAP

criterion in terms of the number of decision agents and their accuracy parameters.

Following the problem setup in [6], a broadly adopted assumption is to consider conditionally independent agents, meaning that there is no communication among them. Under this assumption, the likelihood function of the individual labels breaks down into factors so that the number of parameters of the likelihood function is reduced and the ensemble classification problem becomes more doable. Still, a solution that considers correlated agents can be more appropriate in real applications where agents might be known to share common features. For instance, applications where agents are the same algorithm but with different parameters; agents are trained with the same training data or are specialist from the same school of thought; or even problems with colluding adversaries. The identification of which agents are correlated and which not is out of this work, and interested readers are referred to [13][14].

Existing ensemble classification works that allow for correlated agents are somehow scarce and propose setups with significant differences. A classical approach can be found in [15] where a supervised method to identify a writer by extracting binary correlated features of handwritten text is presented. Indeed, a Neyman-Pearson test is used in which conditional probabilities of all possible observations are estimated using training data. More recently, the usefulness of social learning schemes in ensemble learning, where agents make decisions sequentially based on the observed object and precedent decisions of previous agents, is assessed in [4]. Worthy to mention is [13] where correlation between decision agents is modeled through the use of a set of intermediate latent variables, and the parameters required to build the meta-classifier are estimated using the covariance matrix between pairs of decision agents. Similarly, [14] estimates the agents' error probabilities and provides a strategy to estimate the truth labels even in the presence of colluding adversaries.

In this paper we present a binary unsupervised ensemble classification method suited for correlated decision agents. The labels provided by the agents are modeled using a correlated multivariate Bernoulli distribution as in [16], which is expressed in terms of a set of probability parameters. Following an EM-based algorithm, these parameters are inferred and a final classifier is built. The proposed ensemble classifier is evaluated experimentally both with synthetic and real data sets.

The work is organized as follows. The data model is described in Section II, and Section III presents the ensemble classification method for correlated decision agents. Section IV is devoted to experimental results and finally Section V concludes the work.

Work supported by Spanish Ministry of Economy and Competitiveness ERC funds (TEC2016-77148-C2-1-R and TEC2016-75067-C4-2-R) and Catalan Government funds (2017 SGR 578 AGAUR).

M. Cabrera-Bean (corresponding author) and A. Pagès-Zamora belong to SPCOM Group, Dept. of Signal and Communications Theory, of the Universitat Politècnica de Catalunya BarcelonaTech (UPC), Spain. Contact e-mail: [marga.cabrera@upc.edu](mailto:marga.cabrera@upc.edu) and [alba.pages@upc.edu](mailto:alba.pages@upc.edu).

C. Díaz-Vilor is a master's student at UPC.

## II. PROBLEM SETUP WITH CORRELATED AGENTS

Let's consider  $R$  decision agents who provide binary labels on  $M$  objects and that we want to take a decision on whether each object is positive or negative based on these individual labels. The binary latent variables  $\mathcal{V} := \{v_m\}_{m=1}^M$  denote the ground truth. We also define the set  $\mathcal{Y} := \{\mathbf{y}_1, \dots, \mathbf{y}_M\}$ , where vector  $\mathbf{y}_m$  is an  $R$ -dimensional vector with the binary labels given by the  $R$  agents to classify object  $m^{\text{th}}$ . Each one of these binary labels is modeled as a Bernoulli r.v. We further assume agents can be arranged into  $G$  disjoint groups, so that agents belonging to the same group are correlated whereas those of different groups are not. Labels of correlated agents are modeled using a correlated multivariate Bernoulli distribution as in [16]. We denote by  $R_g$  the number of agents who belong to group  $g$ ; therefore,  $R = \sum_{g=1}^G R_g$  and  $\mathbf{y}_m = [\mathbf{y}_m^1; \dots; \mathbf{y}_m^G]$  where  $\mathbf{y}_m^g$  is an  $R_g \times 1$  vector for  $g = 1, \dots, G$ . Note that  $\mathbf{y}_m^g$  can take  $L_g := 2^{R_g}$  different values, which are denoted by  $\mathbf{z}_l^g$  for  $l = 0, \dots, L_g - 1$ , so that  $\mathbf{z}_l^g$  is the vector form of the decimal to binary conversion of integer  $l$ .

Since decisions of different groups are assumed conditionally independent, the probability mass function (pmf) of the  $R$  binary r.v.'s in  $\mathbf{y}_m$  conditioned on the ground truth is

$$P(\mathbf{y}_m, \mathbf{p}_1) := \Pr(\mathbf{y} = \mathbf{y}_m | v_m = 1) = \prod_{g=1}^G \prod_{l=0}^{L_g-1} (p_{g,l}^1)^{B_l^g(\mathbf{y}_m^g)}$$

$$P(\mathbf{y}_m, \mathbf{p}_0) := \Pr(\mathbf{y} = \mathbf{y}_m | v_m = 0) = \prod_{g=1}^G \prod_{l=0}^{L_g-1} (p_{g,l}^0)^{B_l^g(\mathbf{y}_m^g)} \quad (1)$$

where  $p_{g,l}^n = \Pr(\mathbf{y}_m^g = \mathbf{z}_l^g | v_m = n)$  for  $n \in \{0, 1\}$ ;  $\mathbf{p}_n \in \mathbb{R}^L$  is a vector which entries are equal to the set of parameters  $\{p_{g,l}^n; g = 1, \dots, G; l = 0, \dots, L_g - 1\}$  with  $L = \sum_{g=1}^G L_g$ ; and  $B_l^g(\cdot)$  is the following indicator function

$$B_l^g(\mathbf{y}_m^g) = \begin{cases} 1 & \text{if } \mathbf{y}_m^g = \mathbf{z}_l^g \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

for  $g = 1, \dots, G$ . In order to guarantee that  $P(\mathbf{y}_m, \mathbf{p}_n)$  in (1) are valid pmf's, the parameters  $\{p_{g,l}^n\}_{l=0}^{L_g-1}$  should lie on a probability simplex, i.e.  $0 \leq p_{g,l}^n \leq 1$  and  $\sum_{l=0}^{L_g-1} p_{g,l}^n = 1$ ; for  $n \in \{0, 1\}$ .

It is worthy to note that the distribution in (1) is more general than if we had assumed conditionally independent agents, albeit at the cost of a greater number of parameters, i.e., in our model the set of unknown parameters is  $\theta := \{\pi, \mathbf{p}_1, \mathbf{p}_0\}$  where  $\pi := \Pr(v_m=1)$ . The size of  $\theta$  is  $2L+1$ , where  $L$  increases exponentially with the size of the largest group of correlated agents, instead of  $2R+1$  with conditionally independent agents.

## III. DETECTION ALGORITHM

Given the dataset  $\mathcal{Y}$  and the pmf's in (1), it is well known that the classification criterion that minimizes the error probability is the MAP rule given by

$$\hat{v}_m^{\text{MAP}} = \arg \max_{n \in \{0,1\}} \left( \Pr(v_m = n) P(\mathbf{y}_m, \mathbf{p}_n) \right); m \in \{1, \dots, M\} \quad (3)$$

This rule requires the knowledge of the probability parameters  $\{p_{g,l}^n\}_{l=0}^{L_g-1}$  and prior probabilities  $\Pr(v_m = n)$  for  $g = 1, \dots, G$

and  $n \in \{0, 1\}$ , which can be substituted by the ML estimates. In order to do so, we need the likelihood function of the data set  $\mathcal{Y}$  which, assuming  $M$  independent objects, is equal to a mixture of two  $R \times 1$  correlated Bernoulli distributions, i.e.

$$f(\mathcal{Y}; \theta) = \prod_{m=1}^M \left( \pi P(\mathbf{y}_m, \mathbf{p}_1) + (1 - \pi) P(\mathbf{y}_m, \mathbf{p}_0) \right) \quad (4)$$

However, since there is no closed-form maximization of  $f(\mathcal{Y}; \theta)$ , we resort to the well-known EM algorithm [17].

### A. EM Algorithm for correlated data

The EM algorithm requires the likelihood function of the complete set of r.v.'s, i.e.  $\mathcal{Y}$  and the hidden or latent r.v.'s  $\mathcal{V}$ .

$$f(\mathcal{Y}, \mathcal{V}; \theta) = \prod_{m=1}^M (\pi P(\mathbf{y}_m, \mathbf{p}_1))^{v_m} ((1 - \pi) P(\mathbf{y}_m, \mathbf{p}_0))^{(1-v_m)} \quad (5)$$

The EM algorithm iteratively alternates between the expectation step (E-Step) and the maximization step (M-Step). In the E-Step the following expectation is computed

$$\mathcal{Q}(\theta; \hat{\theta}^t) = \mathbb{E}_{\mathcal{V}} \{ \log(f(\mathcal{Y}, \mathcal{V}; \theta)) | \mathcal{Y}; \hat{\theta}^t \} =$$

$$\sum_{m=1}^M \hat{v}_m^t \log(\pi P(\mathbf{y}_m, \mathbf{p}_1)) + (1 - \hat{v}_m^t) \log((1 - \pi) P(\mathbf{y}_m, \mathbf{p}_0)) \quad (6)$$

where  $\hat{\theta}^t$  denotes the estimate of the parameters at iteration  $t$ , and  $\hat{v}_m^t = \Pr\{v_m = 1 | \mathcal{Y}; \hat{\theta}^t\}$  is the posterior probability of  $v_m$  at iteration  $t$ , equal to

$$\hat{v}_m^t = \frac{\hat{\pi}^t P(\mathbf{y}_m, \hat{\mathbf{p}}_1^t)}{\hat{\pi}^t P(\mathbf{y}_m, \hat{\mathbf{p}}_1^t) + (1 - \hat{\pi}^t) P(\mathbf{y}_m, \hat{\mathbf{p}}_0^t)} \quad (7)$$

In the M-Step, the conditional expectation in (6) is maximized with respect to the parameters  $\theta$ , using Lagrange multipliers to impose the constraints  $\sum_{l=0}^{L_g-1} p_{g,l}^n = 1$ , (see e.g. [18]). Therefore, at iteration  $t$ ,  $\hat{\pi}^{t+1}$  and  $\{\hat{p}_{g,l}^{0,t+1}, \hat{p}_{g,l}^{1,t+1}\}$  for  $g = 1, \dots, G$  and  $l = 0, \dots, L_g - 1$  are computed as follows

$$\hat{\pi}^{t+1} = \frac{1}{M} \sum_{m=1}^M \hat{v}_m^t \quad (8)$$

$$\hat{p}_{g,l}^{1,t+1} = \frac{\sum_{m=1}^M \hat{v}_m^t B_l^g(\mathbf{y}_m^g)}{\sum_{m=1}^M \hat{v}_m^t \sum_{l'=0}^{L_g-1} B_{l'}^g(\mathbf{y}_m^g)} \quad (9)$$

$$\hat{p}_{g,l}^{0,t+1} = \frac{\sum_{m=1}^M (1 - \hat{v}_m^t) B_l^g(\mathbf{y}_m^g)}{\sum_{m=1}^M (1 - \hat{v}_m^t) \sum_{l'=0}^{L_g-1} B_{l'}^g(\mathbf{y}_m^g)} \quad (10)$$

The resulting EM algorithm is summarized in Algorithm 1 and will be referred to as correlated EM (CEM). The final estimate of the parameters is denoted by  $\hat{\theta}^f := \{\hat{\pi}^f, \hat{\mathbf{p}}_0^f, \hat{\mathbf{p}}_1^f\}$ , being  $\hat{p}_{g,l}^{n,f}$  in (9) and (10) the entries of vector  $\hat{\mathbf{p}}_n^f$  for  $n = 0, 1$ ;  $g = 1, \dots, G$  and  $l = 0, \dots, L_g - 1$ . Similarly, the final estimate of the latent variables are denoted by  $\{\hat{v}_m^f\}_{m=1}^M$ .

### B. Decision process

Upon convergence of CEM, a decision on each object is taken based on a binary quantification of variables  $\hat{v}_m^f$ , i.e., if  $\hat{v}_m^f > 0.5$  instance  $m$  is labeled as 1, and 0 otherwise. This is indeed equivalent to apply a MAP criterion in (3) after substituting  $\{\pi, \mathbf{p}_1, \mathbf{p}_0\}$  by  $\{\hat{\pi}^f, \hat{\mathbf{p}}_0^f, \hat{\mathbf{p}}_1^f\}$  of the CEM algorithm.

**Algorithm 1** Correlated EM (CEM)

---

**Input:**  $\{\hat{v}_m^0\}_{m=1}^M, \epsilon, t_{MAX}$   
**Output:**  $\{\hat{v}_m^f\}_{m=1}^M, \hat{\theta}^f$

- 1: **procedure** EM
- 2:      $Q(\theta; \hat{\theta}^0) \leftarrow -\text{inf}; t \leftarrow 0$       $\triangleright$  Initialization
- 3:     **repeat**
- 4:         Compute  $\hat{\pi}^{t+1}$  as in (8)      $\triangleright$  M-Step
- 5:         **for**  $g = 1, \dots, G$  **do**
- 6:             Compute  $\{\hat{p}_{g,l}^{0,t+1}\}_{l=0}^{L_g-1}$  as in (9)
- 7:             Compute  $\{\hat{p}_{g,l}^{1,t+1}\}_{l=0}^{L_g-1}$  as in (10)
- 8:         **end for**
- 9:          $t \leftarrow t + 1; t_f \leftarrow t$
- 10:         Compute  $\{\hat{v}_m^t\}_{m=1}^M$  as in (7)      $\triangleright$  E-Step
- 11:         Compute  $Q(\theta; \hat{\theta}^t)$  as in (6)  $\triangleright$  Check convergence
- 12:         **if**  $Q(\theta; \hat{\theta}^t) - Q(\theta; \hat{\theta}^{t-1}) < \epsilon |Q(\theta; \hat{\theta}^t)|$  **then**
- 13:              $t \leftarrow t_{MAX}$
- 14:         **end if**
- 15:     **until**  $t = t_{MAX}$
- 16:      $\hat{v}_m^f \leftarrow \hat{v}_m^{t_f}, m = 1, \dots, M$
- 17:      $\hat{\theta}^f = \hat{\theta}^{t_f}$
- 18: **end procedure**

---

## IV. NUMERICAL RESULTS

In this section, we support the proposed ensemble classifier with numerical results in MATLAB. To generate the set  $\mathcal{Y} := \{\mathbf{y}_1, \dots, \mathbf{y}_M\}$  of multivariate correlated Bernoulli r.v.'s in the experiments with synthetic data, for  $g = 1 \dots G$  and  $l = 0 \dots L_g - 1$ , we first assign values to the marginal probability parameters  $f_{g,l,1}^n := \Pr(y_m^g(1) = z_l^g(1) | v_m = n)$ , and afterwards to the conditional probability parameters  $f_{g,l,r}^n$  defined as

$$\Pr(y_m^g(r) = z_l^g(r) | y_m^g(1) = z_l^g(1), \dots, y_m^g(r-1) = z_l^g(r-1), v_m = n)$$

for increasing values of  $r$  up to  $R_g$ . Then we compute the probability parameters  $p_{g,l}^n$  as

$$p_{g,l}^n = \Pr(\mathbf{y}_m^g = \mathbf{z}_l^g | v_m = n) = \prod_{r=1}^{R_g} f_{g,l,r}^n \quad (11)$$

for  $n = 0, 1$ ;  $g = 1 \dots G$  and  $l = 0 \dots L_g - 1$ . Additionally, the ground truth set  $\{v_m\}_{m=1}^M$  is generated as independent Bernoulli r.v.'s with parameter  $\pi$ , and the observation vectors  $\mathbf{y}_m$  are obtained according to (1), after substituting the probability parameters computed in (11).

Previous works that generate multivariate Bernoulli distributions impose several consistent relations on the joint probability function [19] or mix different independent Bernoulli trials to correlate them [20]. We use instead the procedure described above to better adjust the correlation between labels of the same group  $g$ . As independence and uncorrelation are equivalent in a multivariable Bernoulli distribution [16], in order to obtain strongly correlated agents we assign values to  $f_{g,l,r}^n$  that depend on the conditional event  $[y_m^g(1) = z_l^g(1), \dots, y_m^g(r-1) = z_l^g(r-1), v_m = n]$  (for instance, we choose a value proportional to integer  $l$ ). The degree of correlation of the data set  $\mathcal{Y}$  will be characterized by  $\{\rho^n\}_{n=0}^1$  defined as

$$\rho^n := \frac{2 \sum_{g=1}^G \sum_{r=1}^{R_g} \sum_{r'=1}^{r-1} |C_g^n(r, r')|}{\sum_{g=1}^G \sum_{r=1}^{R_g} C_g^n(r, r)} \quad (12)$$

TABLE I  
SIMULATION CASES 1 TO 10 FOR  $R = 12$  DECISION AGENTS.

Case	1	2	3	4	5	6	7	8	9	10
$R_1$	1	2	3	4	5	6	7	8	9	10
$\rho$	0.0	0.14	0.29	0.50	0.74	1.02	1.33	1.67	2.04	2.44
$2L + 1$	25	29	35	49	79	141	267	521	1031	2053

with  $C_g^n(r, r') = \mathbb{E}\{[y_m^g(r) - \mathbb{E}\{y_m^g(r)\}][y_m^g(r') - \mathbb{E}\{y_m^g(r')\}] | v_m = n\}$  the covariance between labels  $y_m^g(r)$  and  $y_m^g(r')$ . Indeed, the closer  $\{\rho^0, \rho^1\}$  are to 0, the more independent the data set is.

## A. Results with synthetic data

In all simulation trials we provide  $\rho := \frac{\rho^0 + \rho^1}{2}$  as a measure of correlation degree. Cases 1-10 ( $R = 12$ ) are given in Table I. They consist of 1 group with  $R_1$  correlated agents, and  $R - R_1$  groups with one agent each with an accuracy probability of 0.85. Table I includes parameter  $\rho$  and the number of unknown parameters of CEM,  $2L + 1$ . Note that in Case 1 all agents are uncorrelated. Further, Case 11 ( $\rho = 2.27$  and  $2L + 1 = 237$ ) consists of  $R = 16$  ( $G = 4$ ) with 3 groups with 5 agents each, and one group with a single malicious agent with an accuracy equal to 0.25; and Case 12 ( $\rho = 1.70$  and  $2L + 1 = 67231$ ) consists of  $R = 50$  ( $G = 27$ ) with one group with  $R_1 = 15$  agents, a second one with  $R_2 = 10$  agents and 25 independent agents, some of them are malicious.

In the experiments, CEM is compared to the following methods: MAP criterion as benchmark; Majority Voting (MV); EM algorithm with all uncorrelated agents initialized to MV, denoted by Uncorrelated EM (UEM); and three *state-of-the-art* methods, namely the provably Optimal initialized EM (OEM) [9], the Arbitrary Adversaries based Algorithm (AAA) [14], and the Latent-Spectral Metalearner (LSM) [13]. For the sake of fairness, LSM assumes the group structure is known instead of estimating it as in [13]. Further, the so-called 'Agents' method is the performance averaged over the individual decisions given by the decision agents.

In a first experiment, Cases 1-12 with labels for  $M = 1000$  binary objects with an a priori probability of  $\pi = 0.5$  are considered. Fig. 1 shows the relative error ( $\epsilon_r$ ), defined as the ratio of the number of objects erroneously classified to the total number of objects  $M$ , averaged over  $N = 1000$  independent runs. Indeed  $\epsilon_r = \frac{FP + FN}{M}$ , where  $FP(FN)$  denotes False Positive (False Negative), and the accuracy of a classification method is  $1 - \epsilon_r$ . In these cases, CEM is always superior to the rest of methods and the most similar to MAP. Note that CEM, OEM, LSM and UEM perform the same in Case 1 where all agents are independent.

In a second experiment, we generate  $M = 10 : 10000$  objects for Cases 5 and 10 of Table I, and compute  $\epsilon_r$  averaged along  $N = 1000$  runs. Results in Fig. 2 show that once more, CEM outperforms LSM, UEM, OEM, AAA and MV in case 5, which has less correlated decision agents. Indeed, it approximates MAP performance from  $M = 2000$ . In Case 10, with a higher number of correlated decision agents, CEM is the best from  $M = 500$  onwards, and interestingly CEM performance tends to MAP performance as the number of



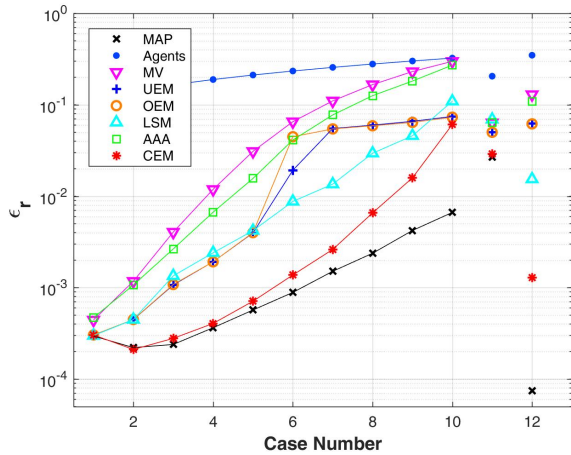


Fig. 1. Averaged relative error  $\epsilon_r$  for different methods and Cases 1 to 12.

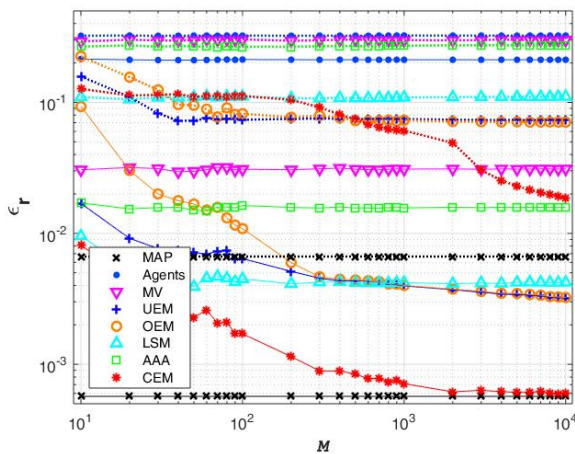


Fig. 2. Averaged relative error  $\epsilon_r$  for Case 5 (continuous lines) and Case 10 (dashed lines) as a function of the number of objects  $M$ .

TABLE II  
MSE MEASURED IN dB FOR CASE 10 AS A FUNCTION OF  $M$

M	10	50	100	500	1000	5000	10000
MSE(dB)	-34.4	-39.8	-41.2	-46.0	-47.8	-57.6	-60.5

objects  $M$  increases. Table II shows the Mean Square Error (MSE) in dB of the parameter estimates versus  $M$  for Case 10. Note that this estimation improves as  $M$  increases, resulting in a decreasing relative error ( $\epsilon_r$ ) in Fig. 2.

First rows of Table III include the classification time of all methods averaging  $N = 1000$  runs for Case 5, with 5 correlated agents; for Case 10 with 10 correlated agents and for  $M = 1000$  and  $M = 10000$ ; and for Case 12, which biggest group includes 15 agents. We can conclude that, even though the number of parameters in CEM increases exponentially with the size of the largest agents' group, still the computational complexity and performance of CEM is competitive with respect to other state-of-the-art methods for sizes of the largest group up to 10 – 15, if necessary at the expense of a higher number of objects  $M$  as in Case 10.

TABLE III  
AVERAGED CLASSIFICATION TIME (ms)

Case	MV	UEM	OEM	LSM	AAA	CEM
5 ( $M = 1000$ )	0.041	23.8	2657.2	126.3	224.9	<b>65.2</b>
10 ( $M = 1000$ )	0.044	25.4	2610.4	121.7	187.0	<b>172.0</b>
10 ( $M = 10000$ )	0.46	<b>255.8</b>	11717.1	1084.0	332.9	<b>2306.3</b>
12 ( $M = 1000$ )	0.096	96.7	5528.2	461.2	23632.3	<b>2787.5</b>
Spam ( $M = 1150$ )	2.63	39,41	5115.5	268.4	663.0	<b>112.4</b>
Star ( $M = 4474$ )	9.88	149.6	12072.7	<b>717.6</b>	712.1	<b>247.1</b>

TABLE IV  
AVERAGED  $\epsilon_r$  FROM REAL DATA, GROUP STRUCTURE: [6, 6, 6]

Dataset	Agents	MV	UEM	OEM	LSM	AAA	CEM
Spam ( $M=1150$ )	0.115	0.081	0.081	0.082	0.080	0.081	<b>0.078</b>
Star ( $M=4474$ )	0.025	0.019	0.025	0.022	<b>0.018</b>	0.019	<b>0.018</b>

### B. Results with real datasets

Two UCI repository real datasets from archive.ics.uci.edu have been selected to validate the CEM algorithm. The Spam dataset consists of  $N = 4601$  56-featured instances of an e-mail text that can be spam. The HTRU2 dataset consists of  $N = 17898$  8-featured instances of a star observation that potentially can be a pulsar star. For each one of the datasets,  $R = 18$  classifiers are checked using  $G = 3$  groups with  $R_1 = 6$  neural networks,  $R_2 = 6$  support vector machines and  $R_3 = 6$  decision tree classifiers, with a total of  $2L + 1 = 385$  unknown parameters. Each individual classifier is firstly trained on a randomly chosen subset. Then, a separate single independent test dataset with  $M = 1150$  instances for Spam and  $M = 4474$  instances for Star is used to obtain the set  $\mathcal{Y} := \{\mathbf{y}_1, \dots, \mathbf{y}_M\}$  with  $\mathbf{y}_m \in \mathbb{R}^{18}$ . Table IV shows the relative error  $\epsilon_r$  obtained averaged over 1000 realizations, and last rows in Table III show the classification time spent in these experiments. CEM attains the best performance for Spam case, and LSM and CEM are the best ones for the Stars, with less computational cost for CEM.

## V. CONCLUSIONS

A new unsupervised ensemble classification method for optimally combining labels provided by correlated decision agents is presented. As shown with numerical results using synthetic and real datasets, the proposed method achieves a higher classification accuracy than other classical and state-of-the-art algorithms for medium-size groups of correlated agents. Moreover, even though the number of parameters of the proposed method scales exponentially with the number of correlated agents, the computational time is still competitive compared to other state-of-the-art methods and accuracy improves as the number of objects to classify increases. Further work includes the generalization to multiple classes; dealing with missing labels; on-line implementations to estimate agents' performance, especially if they are going to be used later on; and identification of correlated groups.

## REFERENCES

- [1] V. C. Raykar, S. Yu, L. H. Zhao, G. H. Valadez, C. Florin, L. Bogoni, and L. Moy, "Learning from crowds," *Journal of Machine Learning Research*, vol. 11, no. Apr., pp. 1297–1322, 2010.
- [2] M. Micsinai, F. Parisi, F. Strino, P. Asp, B. D. Dynlacht, and Y. Kluger, "Picking chip-seq peak detectors for analyzing chromatin modification experiments," *Nucleic acids research*, vol. 40, no. 9, pp. e70–e70, 2012.
- [3] A. Pagès-Zamora, M. Cabrera-Bean, and C. Díaz-Vilor, "Unsupervised online clustering and detection algorithms using crowdsourced data for malaria diagnosis," *Pattern Recognition*, vol. 86, pp. 209–223, 2019.
- [4] J. B. Rhim and V. K. Goyal, "Distributed hypothesis testing with social learning and symmetric fusion," *IEEE Trans. on Signal Processing*, vol. 62, no. 23, pp. 6298–6308, 2014.
- [5] M. Usman and K. Insoo, "Sensor network-based spectrum sensing for cognitive radio network," in *Int. Conf. on Intelligent Systems Engineering (ICISE)*. IEEE, 2016, pp. 19–25.
- [6] A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the em algorithm," *Applied statistics*, pp. 20–28, 1979.
- [7] F. Parisi, F. Strino, B. Nadler, and Y. Kluger, "Ranking and combining multiple predictors without labeled data," *Proc. of the National Academy of Sciences*, vol. 111, no. 4, pp. 1253–1258, 2014.
- [8] A. Jaffe, B. Nadler, and Y. Kluger, "Estimating the accuracies of multiple classifiers without labeled data," in *Artificial Intelligence and Statistics*, 2015, pp. 407–415.
- [9] Y. Zhang, X. Chen, D. Zhou, and M. I. Jordan, "Spectral methods meet em: A provably optimal algorithm for crowdsourcing," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 3537–3580, 2016.
- [10] P. Traganitis, A. Pagès-Zamora, and G. B. Giannakis, "Blind multiclass ensemble classification," *IEEE Trans. on Signal Processing*, 2018.
- [11] H. Li and B. Yu, "Error rate bounds and iterative weighted majority voting for crowdsourcing," *arXiv preprint arXiv:1411.4086*, 2014.
- [12] M. Cabrera-Bean, C. Díaz-Vilor, and J. Vidal, "Impact of noisy annotators' reliability in a crowdsourcing system performance," in *European Signal Processing Conference (EUSIPCO)*. EURASIP, 2016, pp. 2005–2009.
- [13] A. Jaffe, E. Fetaya, B. Nadler, T. Jiang, and Y. Kluger, "Unsupervised ensemble learning with dependent classifiers," in *Artificial Intelligence and Statistics*, 2016, pp. 351–360.
- [14] M. Kleindessner and P. Awasthi, "Crowdsourcing with arbitrary adversaries," in *Int. Conf. on Machine Learning*, 2018, pp. 2713–2722.
- [15] E. N. Zois and V. Anastassopoulos, "Morphological waveform coding for writer identification," *Pattern Recognition*, vol. 33, no. 3, pp. 385–398, 2000.
- [16] B. Dai, S. Ding, and G. Wahba, "Multivariate bernoulli distribution," *Bernoulli*, vol. 19, no. 4, pp. 1465–1483, 09 2013.
- [17] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.
- [18] C. M. Bishop, *Pattern recognition in Machine Learning*. Cambridge, UK: Springer Science, 2006.
- [19] M. Hisakado, K. Kitsukawa, and S. Mori, "Correlated binomial models and correlation structures," *Journal of Physics A: Mathematical and General*, vol. 39, no. 50, p. 15365, 2006.
- [20] R. Pires M and C. A. Diniz, "Correlated binomial regression models," *Computational Statistics and Data Analysis*, vol. 56, no. 8, pp. 2513–2525, 2012.