Co-training with Credal Models

Yann Soullard(✉), Sébastien Destercke, and Indira Thouvenin

Sorbonne University, Université de Technologie de Compiègne,
CNRS UMR 7253 Heudiasyc, CS 60 319, 60 203 Compiègne Cedex, France
{yann.soullard,sebastien.destercke,indira.thouvenin}@hds.utc.fr

Abstract. So-called credal classifiers offer an interesting approach when the reliability or robustness of predictions have to be guaranteed. Through the use of convex probability sets, they can select multiple classes as prediction when information is insufficient and predict a unique class only when the available information is rich enough. The goal of this paper is to explore whether this particular feature can be used advantageously in the setting of co-training, in which a classifier strengthen another one by feeding it with new labeled data. We propose several co-training strategies to exploit the potential indeterminacy of credal classifiers and test them on several UCI datasets. We then compare the best strategy to the standard co-training process to check its efficiency.

Keywords: Co-training · Imprecise probabilities · Semi-supervised learning · Ensemble models

1 Introduction

There are many application fields (gesture, human activity, finance, ...) where extracting numerous unlabeled data is easy, but where labeling them reliably require costly human efforts or an expertise that may be rare and expensive. In this case, getting a large labeled dataset is not possible, making the task of training an efficient classifier from labeled data alone difficult. The general goal of semi-supervised learning techniques [1,7,28] is to solve this issue by exploiting the information contained in unlabeled data. It includes different approaches such as the adaptation of training criteria [13,14,16], active learning methods [18] and co-training-like approaches [6,19,22].

In this paper, we focus on the co-training framework. This approach aims at training two classifiers in parallel, and each model then attempts to strengthen the other by labeling a selection of unlabeled data. We will call *trainer* the classifier providing new labeled instances and *learner* the classifier using it as new training data. In the standard co-training approach [6,22], the trainer provides to the learner the data about which it gets the most confident labels. However, those labels are predicted with high confidence by the trainer but it is not guaranteed that the new labeled instances will be informative for the learner, in the sense that it may not help him to improve its accuracy.

F. Schwenker et al. (Eds.): ANNPR 2016, LNAI 9896, pp. 92–104, 2016.
DOI: 10.1007/978-3-319-46182-3_8

To solve this issue, we propose a new co-training approach using credal classifiers. Such classifiers, through the use of convex sets of probabilities, can predict a set of labels when training data are insufficiently conclusive. It means they will produce a single label as prediction only when the information is enough (i.e., when the probability set is small enough). The basic idea of our approach is to select as potential new training data for the learner those instances for which the (credal) trainer has predicted a single label and the learner multiple ones.

2 Co-training Framework

We assume that samples are elements of a space $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ is the input space and $\mathcal{Y}$ the output space of classes. In a co-training setting, it is assumed that the input space $\mathcal{X}$ can be split into two different views $\mathcal{X}_1 \times \mathcal{X}_2$ and that classifiers can be learned from each of those views. That is, an instance $\mathbf{x} \in \mathcal{X}$ can be split into a couple $(\mathbf{x}_1, \mathbf{x}_2)$ with $\mathbf{x}_j \in \mathcal{X}_j$. In the first works introducing co-training [6,22], it is assumed that each view is sufficient for a correct classification and is conditionally independent to the other given the class label. However, it has been shown that co-training can also be an efficient semi-supervised techniques when the views are insufficient or when labels are noisy [24]. In addition, many studies provide theoretical results on co-training: [4] shows that the assumption of conditional independence can be relaxed to some extent; [11] gives some theoretical justifications of the co-training algorithm, providing a bound on the generalization error that is related on the empirical agreement between the two classifiers; [23] analyzes the sufficient and necessary condition for co-training to succeed through a graph view of co-training. Besides, in [26], the authors propose to estimate the labeling confidence using data editing techniques, allowing especially to identify an appropriate number of predicted examples to pass on to the other classifier.

We will denote by $\mathcal{L}_j$ the set of labeled examples from which a model h_j is learned i.e. $\mathcal{L}_j = \{(\mathbf{x}_j^{(i)}, y^{(i)}), i \in \{1, ..., m_j\}\}$ where $\mathbf{x}_j^{(i)}$ is the j^{th} view of the instance $\mathbf{x}^{(i)}$ and m_j denotes the number of labeled examples in $\mathcal{L}_j$. Co-training starts with a (usually small) common set of labeled examples, i.e. $\mathcal{L}_1 = \{(\mathbf{x}_1^{(i)}, y^{(i)}), i \in \{1, ..., m\}\}$ and $\mathcal{L}_2 = \{(\mathbf{x}_2^{(i)}, y^{(i)}), i \in \{1, ..., m\}\}$, and a pool of unlabeled examples $\mathcal{U} = \{(\mathbf{x}_1^{(i)}, \mathbf{x}_2^{(i)}), i \in \{m+1, ..., n\}\}$. Based on previous works of [22], the standard co-training method that we will adapt to imprecise probability setting goes as follow: at each step of the co-training process, two classifiers $h_j : \mathcal{X}_j \to \mathcal{Y}$ ($j \in \{1, 2\}$) are learned from the learning set $\mathcal{L}_j$. We also assume that classifier h_j uses an estimated conditional probability distribution $p_j(\cdot|\mathbf{x}) : \mathcal{Y} \to [0, 1]$ to take its decision, i.e.

$$h_j(\mathbf{x}_j) = \arg\max_{y \in \mathcal{Y}} p_j(y|\mathbf{x}_j). \tag{1}$$

The n_u examples labeled by h_j with the most confidence (the highest probability, in our case) are then added to the set $\mathcal{L}_k$ of training examples of $h_k, k \neq j$ and

Input: h_1, h_2 learned from $\mathcal{L}_1, \mathcal{L}_2$, set $\mathcal{U}$, number n_u of added learning samples;
Output: Updated sets $\mathcal{L}_1, \mathcal{L}_2$
n=1;
repeat
  set $\mathbf{x}^* = \arg\max_{\mathbf{x}^{(i)} \in \mathcal{U}} p_1(h_1(\mathbf{x}_1^{(i)})|\mathbf{x}_1^{(i)})$;
  $\mathcal{L}_2 \leftarrow \mathcal{L}_2 \cup (\mathbf{x}^*, h_1(\mathbf{x}^*))$;
  set $\tilde{\mathbf{x}}^* = \arg\max_{\mathbf{x}^{(i)} \in \mathcal{U}} p_2(h_2(\mathbf{x}_2^{(i)})|\mathbf{x}_2^{(i)})$;
  $\mathcal{L}_1 \leftarrow \mathcal{L}_1 \cup (\tilde{\mathbf{x}}^*, h_2(\tilde{\mathbf{x}}^*))$;
  $\mathcal{U} \leftarrow \mathcal{U} \setminus \{\mathbf{x}^*, \tilde{\mathbf{x}}^*\}$;
  $n \leftarrow n + 1$;
until $n = n_u$;

Algorithm 1. Standard co-training procedure

removed from $\mathcal{U}$. One iteration of this process is summarized by Algorithm 1. The procedure is then iterated a number of pre-defined times.

Note that co-training can also be used with two different classifiers using the same view $\mathcal{X}$. [22] provide a theoretical analysis demonstrating that two learners can be improved in such a procedure provided they have a large difference. This view is taken further by [27] that studies ensemble learning (with more than two classifiers) in a semi-supervised framework.

3 Basics of Credal Models

We introduce the basic elements we need about imprecise probabilities. Interested readers are referred to [3] for further details.

3.1 Imprecise Probabilities and Decision

Let $\mathcal{Y}$ be a finite space (e.g., of classes) and $\Sigma_\mathcal{Y}$ be the set of all probability mass functions over $\mathcal{Y}$. In imprecise probability theory, the uncertainty about a variable Y is described by a convex set $\mathcal{P} \subseteq \Sigma_\mathcal{Y}$ of probabilities which is usually called *credal set*. When this convex set depends on some input data $\mathbf{x}$, as in classification problems, we will denote it by $\mathcal{P}_\mathbf{x}$. Given $\mathbf{x}$, lower and upper probabilities of elements of $y \in \mathcal{Y}$ can then be defined as:

$$\underline{p}(y|\mathbf{x}) = \inf_{p(\cdot|\mathbf{x}) \in \mathcal{P}_\mathbf{x}} p(y|\mathbf{x}) \quad \text{and} \quad \overline{p}(y|\mathbf{x}) = \sup_{p(\cdot|\mathbf{x}) \in \mathcal{P}_\mathbf{x}} p(y|\mathbf{x}) \tag{2}$$

and when $\mathcal{P}_\mathbf{x}$ is reduced to a singleton, we retrieve the probabilistic case where $\underline{p}(y|\mathbf{x}) = \overline{p}(y|\mathbf{x})$ for any $y \in \mathcal{Y}$. Probabilities of events and expected values can be extended in the same way, by considering boundary values.

Many different ways have been proposed for extending the classical decision criterion $h(\mathbf{x}) = \arg\max_{y \in \mathcal{Y}} p(y|\mathbf{x})$ within imprecise probability theory [20]. Some of them, such as the maximin that replaces $p(y|\mathbf{x})$ by $\underline{p}(y|\mathbf{x})$ in Eq. (1), also produce unique predictions. Others, that we will use here, can produce sets

of possible predictions, the size of the set reflecting the lack of information. Such a decision rule is then a mapping[1] $H : \mathcal{X} \rightarrow 2^{\mathcal{Y}}$ from the input set to the power set of classes.

In this paper, we will focus on two of the most popular decision rules that are the so-called *interval dominance* and the *maximality* criteria. The *interval dominance* decision rule is defined as:

$$H(\mathbf{x}) = \{y \in \mathcal{Y} | \quad \nexists\, y' s.t. \quad \underline{p}(y'|\mathbf{x}) > \overline{p}(y|\mathbf{x})\}. \tag{3}$$

The idea of this rule is that a class is rejected as a possible prediction when its upper bound of probability is lower than the lower bound of at least one other class. However, this means that two classes y, y' may be compared according to different precise probabilities within $\mathcal{P}_{\mathbf{x}}$ (as their boundary probabilities can be obtained at different points). This is not the case of the *maximality* decision rule, that rejects a class which is certainly less probable (according to any $p \in \mathcal{P}_{\mathbf{x}}$) than the others:

$$H(\mathbf{x}) = \{y \in \mathcal{Y} | \quad \nexists\, y' s.t. \quad p(y'|\mathbf{x}) > p(y|\mathbf{x}) \forall\, p(\cdot|\mathbf{x}) \in \mathcal{P}_{\mathbf{x}}\}. \tag{4}$$

Those decision rules are reduced to (1) in a precise framework, as it consists in choosing the top element of the order induced by the probability weights. The set (4) can be computed in the following way: starting from the whole set $\mathcal{Y}$, if the value

$$\inf_{p(\cdot|\mathbf{x}) \in \mathcal{P}_{\mathbf{x}}} p(y|\mathbf{x}) - p(y'|\mathbf{x}) \tag{5}$$

is strictly positive for a given pair of classes y, y', then y' can be removed from $H(\mathbf{x})$, since if (5) is positive, $p(y|\mathbf{x})$ is strictly greater than $p(y'|\mathbf{x})$ for all $p(\cdot|\mathbf{x})$. The set (4) can then be obtained by iterating this procedure over all pairs of classes, removing those that are not optimal. Note that this approach will produce set predictions both in case of ambiguity ($\mathcal{P}_{\mathbf{x}}$ may be small, but may contain probabilities whose higher values are similar) and of lack of information ($\mathcal{P}_{\mathbf{x}}$ is large because few training data were available), and will therefore produce precise predictions (the cardinality $|H(\mathbf{x})| = 1$) only when being very confident. It should be noted that the set produced by Eq. (3) will always include (hence will be more precise) the one produced by Eq. (4).

3.2 Learning Credal Models

A common way to learn credal models is to extend Bayesian models by considering sets of priors. After learning, they provide a set of posterior probabilities which converges towards a single probability when the training dataset size increases. The most well-known example is the naive credal classifier [25], that extends the naive Bayes classifier by learning set of conditional probabilities, using the so-called imprecise Dirichlet model [5].

[1] We use capital letter to denote the fact that the returned prediction may be a set of classes.

Some works combine sets of credal classifiers to extend popular techniques such as Bayesian model averaging [8], binary decomposition [12] or boosting [21], as well as there are some preliminary works that exploit credal approaches in semi-supervised settings [2,17]. However, we are not aware of any work trying to exploit credal models to mutually enrich sets of classifiers.

4 Co-training with Credal Models

We propose a new co-training approach based on credal models that extends the standard co-training framework recalled in Sect. 2. We will refer to it as *credal co-training*.

4.1 Motivation and Idea

Working with a small labeled training set may cause several issues in semi-supervised learning, one major issue being the possible bias resulting from few samples [15,27]. Second, in the standard co-training framework recalled in Sect. 2, there is no guarantee that the data selected by the trainer will actually be useful to the learner, in the sense that the learner may already be quite accurate on those data.

We think our proposal tackles, at least partially, both problems. Working with sets of priors and with sets of probabilities whose sizes depend on the number of training data is a way to be less affected by possible bias. Second, the fact that predictions are set-valued can help to identify on which data the trainer is really confident (those for whose it makes a unique prediction) and the learner is not (those for which it makes set-valued prediction). Based on this distinction, our approach consists in modifying Algorithm 1 in two different aspects:

- Select data from a subset of $\mathcal{U}$, denoted $\mathcal{S}_{H_i \to H_j} \subseteq \mathcal{U}$, corresponding to data for which H_i is confident and H_j is not (Sect. 4.2).
- Adapt the notion of confidence to imprecise probabilities, to choose specific data from $\mathcal{S}_{H_i \to H_j}$ (Sect. 4.3).

The resulting co-training process, that we will detail in the next sections, is illustrated in Fig. 1.

4.2 Dataset Selection Among the Unlabeled Instances

In this section, we define the set $\mathcal{S}_{H_i \to H_j}$ containing the pool of unlabeled data that may be labeled by the trainer H_i for the learner H_j. The idea is that it should contain data for which H_i (the trainer) is confident and H_j (the learner) is not. We denote $S^c_{H_i}$ the dataset on which H_i provides precise classifications and $S^u_{H_j}$ the one on which H_j provides indeterminate predictions. The set $\mathcal{S}_{H_i \to H_j}$ is defined as follow:

$$S_{H_i \to H_j} = S^c_{H_i} \cap S^u_{H_j} := \{\mathbf{x} \in \mathcal{U} \mid |H_i(\mathbf{x})| = 1 \wedge |H_j(\mathbf{x})| > 1\} \tag{6}$$

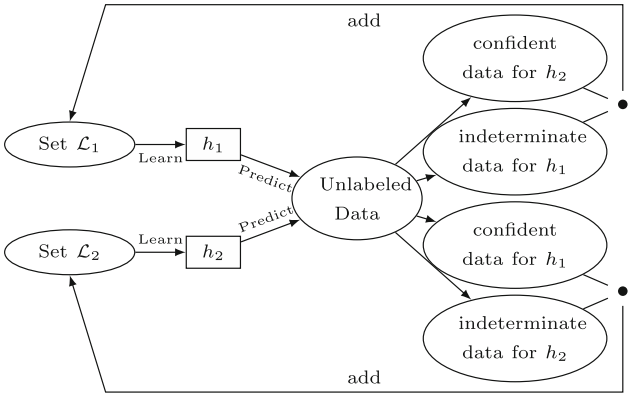

Fig. 1. Co-training with credal models.

It contains the unlabeled data for which H_i predicts a unique value and H_j multiple ones. As this set may be empty, we reduce the imprecision of the trainer so that it converges towards the precise framework. This guarantees that the trainer have confident instances that may be selected for labeling, but this may reduce the trainer confidence about its predictions. We will discuss in details this strategy in our experimentations (see Sect. 5.2). Note that when the learner H_j is confident on $\mathcal{U}$, the pool of unlabeled data is reduced to the instances for which the trainer is confident, i.e. $S_{H_i \to H_j} = S^c_{H_i}$.

4.3 Data Selection for Labeling

Given a set of unlabeled data $\mathcal{U}$ and two models H_1 and H_2 learned from $\mathcal{L}_1$ and $\mathcal{L}_2$ respectively, we define several strategies for selecting a data in the unlabeled dataset $S_{H_i \to H_j}$. A first one, that we call *uncertain strategy*, consists in choosing the data for which the learner is the less confident:

$$\mathbf{x}^\star = \underset{\mathbf{x} \in S_{H_i \to H_j}}{\arg\max} \sum_{y \in \mathcal{Y}} \left(\overline{p}_j(y|\mathbf{x}) - \underline{p}_j(y|\mathbf{x}) \right) \tag{7}$$

where $\underline{p}_j$ and $\overline{p}_j$ are the lower and upper probabilities induced from $\mathcal{L}_j$. We therefore replace Line 3 of Algorithm 1 by Eq. (7) with $i = 1, j = 2$, and Line 5 likewise with $i = 2, j = 1$ (the same will apply to all strategies). The idea of the uncertain strategy is that H_j gains information where it is the least informed. We refine this strategy by focusing on the classes predicted by the learner H_j and on data for which H_j is the most uncertain, what we call the *indeterminate strategy*:

$$\mathbf{x}^\star = \underset{\mathbf{x} \in S_{H_i \to H_j}}{\arg\max} \left(\mathbf{1}_{|H_j(\mathbf{x})| = \max_{\mathbf{x}' \in \mathcal{U}} |H_j(\mathbf{x}')|} \times \sum_{y \in H_j(\mathbf{x})} \left(\overline{p}_j(y|\mathbf{x}) - \underline{p}_j(y|\mathbf{x}) \right) \right) \tag{8}$$

Those two strategies are not extensions of the standard framework, in the sense that we do not retrieve Algorithm 1 when $\underline{p} = \overline{p}$. The next strategies are

such extensions, and are based on probability bounds. The first one, that we call *optimistic strategy*, consists in selecting the data with the highest upper probability among the trainer predictions:

$$\mathbf{x}^\star = \underset{\mathbf{x}\in S_{H_i\to H_j}}{\arg\max} \max_{y\in H_i(\mathbf{x})} \overline{p}_i(y|\mathbf{x}) \tag{9}$$

In a similar manner, the *pessimistic strategy* selects the data with the highest lower probability:

$$\mathbf{x}^\star = \underset{\mathbf{x}\in S_{H_i\to H_j}}{\arg\max} \max_{y\in H_i(\mathbf{x})} \underline{p}_i(y|\mathbf{x}) \tag{10}$$

The last strategy, called *median*, consists in selecting the unlabeled data with the highest median value in the intervals of probability:

$$\mathbf{x}^\star = \underset{\mathbf{x}\in S_{H_i\to H_j}}{\arg\max} \max_{y\in H_i(\mathbf{x})} \frac{1}{2}\big(\underline{p}_i(y|\mathbf{x}) + \overline{p}_i(y|\mathbf{x})\big). \tag{11}$$

Those last strategies correspond to common decision criteria used in imprecise probability theory (and robust methods in general), that are respectively the maximax, maximin and Hurwicz criteria [20]. As in the standard framework, these strategies aim at giving informative samples to the learner, but intend to exploit the robustness of credal models.

5 Experimentations

We experiment our proposed approach on various UCI datasets described in Table 1. We use a Naive Credal Classifier [9,10,25] to test our approach. For each dataset, we split the feature set in two distinct parts where the first half serve as the first view and the second half as the second view. Although we have no guarantee that the two views are sufficient, recent results [24] suggest that co-training can also work in such a setting. Thus, we expect that our approach will be able to improve the supervised case and that it will overcome the standard co-training framework.

5.1 Naive Credal Classifier (NCC)

The Naive Credal Classifier (NCC) [10,25] is an extension of Naive Bayes Classifier (NBC) to imprecise probabilities. Let $\mathbf{x} = (x_1, .., x_K)$ be an instance and let us denote $\mathcal{P}_\mathcal{Y}$ a credal set of prior distributions $p(y)$ on the classes, and $\mathcal{P}^y_{x_k}$ a credal set of conditional distributions $p(x_k|y)$. The model is characterized by a set of joint probability distributions $p(y, \mathbf{x})$ which satisfies the assumption that, for a given class y, the value of a feature x_k for $k \in \{1, ..., K\}$ is independent of the value of any other feature. According to this assumption, the model is defined as follow:

$$p(y, \mathbf{x}) = p(y, x_1, .., x_K) = p(y) \prod_{k=1}^{K} p(x_k|y) \tag{12}$$

Table 1. UCI datasets, with the number of data, of features and of classes.

Name	#Samples	#Features	#Classes
Diabetes	768	8	2
Haberman	306	3	2
Ionosphere	351	34	2
Iris	150	4	3
KDD synth.	600	60	6
Kr vs. kp	3196	36	2
Mfeat morph.	2000	6	10
Opdigits	5620	64	10
Page-blocks	5473	10	5
Segment	2310	19	7
Spambase	4601	57	2
Wine	178	13	3

with $p(y) \in \mathcal{P}_{\mathcal{Y}}$ and $p(x_k|y) \in \mathcal{P}^y_{x_k}$ for $k \in \{1, ..., K\}$ and $y \in \mathcal{Y}$. This results in a set of posterior probabilities. Conditional credal sets $\mathcal{P}^y_{x_k}$ are typically learned using the Imprecise Dirichlet model [5,25] provides an efficient procedure to compute the sets (3) and (4) respectively based on the interval dominance and maximality decision rules, we refer to this work for details.

5.2 Comparison of Data Selection Strategies

We first compare the various data selection strategies defined in Sect. 4.3. The five strategies are compared to the supervised framework. The supervised case provides the initial performances on each view obtained without using $\mathcal{U}$. The co-training is performed during 50 training iterations and, at each iteration, one instance is labeled per model. Table 2 shows some results on a 10-fold cross validation where, for each fold, 10 % of the data are used for the test, 40 % of them are labeled instances used for training the models[2] and the rest is considered to be unlabeled and may be selected by the models using one of the strategies. As defined by [25], we use a NCC hyper-parameter for controlling at which speed the credal set converge towards a unique probability. In our experiments, this hyper-parameter, called *s value* (see [25] for details) is equal to 5 for the trainer and 2 for the learner. At each co-training iteration, we decrease the s value of the trainer if the pool of unlabeled data (6) defined in Sect. 4.2 is empty until $s = 0$ (the precise setting). If the learner has no indeterminate predictions (i.e. $S^u_{h_L} = \emptyset$), a data is selected in the pool of data for which the trainer is confident.

[2] We use 40 % of them as labeled instances to get a compromise between having a large part of data as unlabeled and having a sufficiently large labeled dataset to reduce the sampling bias.

Table 2. Performances of the uncertainty (*UNC*), indeterminate (*IND*), optimistic (*OPT*), pessimistic (*PES*) and median (*MED*) strategies of the credal co-training (with maximality) compared to the supervised setting (*SUP*).

	FIRST MODEL						SECOND MODEL					
DATASET	SUP.	UNC.	IND.	OPT.	PES.	MED.	SUP.	UNC.	IND.	OPT.	PES.	MED.
Diabetes	73.9%	74.0%	74.0%	**74.2%**	73.7%	73.7%	70.5%	**72.5%**	**72.5%**	71.9%	72.1%	72.4%
Ionosphere	89.8%	**90.1%**	**90.1%**	**90.1%**	**90.1%**	89.8%	79.9%	81.5%	81.5%	**82.9%**	81.7%	81.7%
Iris	71.3%	76.7%	75.3%	**77.3%**	76.0%	75.3%	92.7%	93.3%	92.7%	**93.3%**	**93.3%**	**93.3%**
KDD synthetic control	75.2%	77.3%	**77.7%**	77.3%	77.5%	77.5%	84.3%	85.3%	85.2%	**85.7%**	85.2%	85.2%
Page-blocks	**92.9%**	92.2%	91.4%	**92.9%**	92.3%	92.3%	**86.8%**	85.5%	85.5%	86.1%	86.1%	86.2%
Segment	57.7%	58.5%	**58.7%**	58.2%	58.5%	58.5%	81.8%	**83.3%**	82.5%	82.5%	81.6%	81.5%
Spambase	86.4%	**86.6%**	**86.6%**	**86.6%**	**86.6%**	**86.6%**	84.5%	**84.6%**	**84.6%**	84.4%	84.5%	84.5%
Wine	82.0%	**88.5%**	**88.5%**	**88.5%**	87.9%	**88.5%**	93.2%	95.0%	95.4%	**96.3%**	**96.3%**	**96.3%**

Results of Table 2 show that our approach generally improves the supervised framework. This confirms the interest of exploiting unlabeled data by a co-training process as the one we propose. Both the *uncertain* and *indeterminate* strategies provide similar results: they often get robust performances but when the credal co-training is not relevant, i.e. when the credal co-training is weaker than the simple supervised setting whatever the selection method, they are less efficient than the other strategies.

In contrast, other strategies (*optimistic*, *pessimistic* and *median*) seems to be more robust. They have close performances but the *optimistic* strategy presents overall the best performances. As using an optimistic strategy is common in semi-supervised setting, we will use only the optimistic strategy in the next experiments comparing our approach to the standard one (Sect. 5.3).

5.3 Comparison with Standard Co-training

Having defined our data selection strategy for credal co-training, we now compare it to the standard co-training strategy recalled in Sect. 2 to confirm that it performs at least as well as this latter one (Table 3). To do so, we start from the same initial data sets and run the two co-training settings in parallel. Once those co-training processes are done, the final learning sets are used to learn standard Naive Bayes Classifiers (producing determinate predictions), whose their usual accuracies are then compared. Thus, credal models are used in the credal co-training phase, but not to obtain a final predictive model.

The experimental setting is the same as in Sect. 5.2. Here, STD stands for the standard method, while PACC stands for the precise accuracy of the Naive Bayes Classifier learned after the credal co-training process. We experiment the credal co-training with the *interval dominance* (INT) and *maximality* (MAX) decision rules.

The co-training with credal models almost always improve the supervised case and it is generally better than the standard co-training framework. A standard Wilcoxon test comparing the credal co-training performances with those of the standard co-training give a p-value of 0.13 for PACC.-INT. (interval

Table 3. Comparison of Naive Bayes Classifier performances after a credal co-training process (PREC-ACC), a standard co-training (STD) and a supervised training (SUP).

	FIRST MODEL				SECOND MODEL			
DATASET	SUP.	STD	PACC.-INT.	PACC.-MAX.	SUP.	STD	PACC.-INT.	PACC.-MAX.
Diabetes	73.9% ± 3.9	73.7% ± 4.3	**74.2%** ± 4.7	**74.2%** ± 4.7	70.5% ± 3.4	**72.1%** ± 4.7	71.9% ± 3.3	71.9% ± 3.3
Haberman	**73.5%** ± 0.5	**73.5%** ± 0.5	**73.5%** ± 0.5	**73.5%** ± 0.5	74.0% ± 6.7	**74.5%** ± 3.0	74.2% ± 1.4	74.2% ± 1.4
Ionosphere	89.8% ± 5.9	**90.2%** ± 6.1	90.1% ± 5.3	90.1% ± 5.3	79.9% ± 7.8	82.3% ± 6.6	**82.9%** ± 6.1	**82.9%** ± 6.1
Iris	71.3% ± 11.2	75.3% ± 10.8	74.7% ± 11.1	**77.3%** ± 10.4	92.7% ± 5.5	93.3% ± 6.0	**94.0%** ± 5.5	93.3% ± 5.2
KDD synthetic control	75.2% ± 4.9	74.8% ± 5.4	**77.3%** ± 5.3	**77.3%** ± 5.3	84.3% ± 6.6	82.0% ± 6.0	**85.7%** ± 5.4	**85.7%** ± 5.4
Kr vs kp	**70.6%** ± 2.1	70.0% ± 2.3	70.2% ± 2.4	70.2% ± 2.4	80.9% ± 1.3	**81.1%** ± 1.2	80.8% ± 1.9	80.8% ± 1.9
Mfeat morphological	**46.3%** ± 1.2	46.2% ± 1.5	45.8% ± 0.9	46.0% ± 1.0	44.2% ± 4.2	43.2% ± 4.0	44.3% ± 3.6	**44.4%** ± 3.9
Optdigits	78.8% ± 1.9	78.8% ± 1.8	**79.2%** ± 1.9	79.1% ± 1.9	78.0% ± 2.5	78.1% ± 2.4	**78.3%** ± 2.3	78.2% ± 2.3
Page-blocks	**92.9%** ± 1.2	**92.9%** ± 1.2	92.3% ± 1.1	**92.9%** ± 1.0	**86.8%** ± 1.5	86.7% ± 1.4	86.3% ± 1.4	86.1% ± 1.5
Segment	57.7% ± 2.8	58.1% ± 2.8	**58.2%** ± 2.7	**58.2%** ± 2.8	81.8% ± 2.0	82.0% ± 1.8	**82.6%** ± 2.0	82.5% ± 1.9
Spambase	86.4% ± 1.0	**86.6%** ± 1.0	86.5% ± 1.2	**86.6%** ± 1.1	**84.5%** ± 1.4	84.1% ± 1.3	84.4% ± 1.3	84.4% ± 1.3
Wine	82.0% ± 9.0	86.0% ± 6.1	**88.5%** ± 6.0	**88.5%** ± 6.0	93.2% ± 5.7	94.5% ± 4.9	95.4% ± 4.5	**96.3%** ± 3.0

dominance) and of $0,07$ for PACC.-MAX., indicating that credal co-training with maximality would be statistically higher if the significance were set to 0.1 threshold. Moreover, there are few cases in which co-training is harmful to the performances: this is probably due to too insufficient view, in which case co-training performances may suffer of label noise or sampling bias as mentioned by [24].

5.4 Behaviours of Credal Models During the Co-training Iterations

Having confirmed the interest of a credal co-training approach, we now examine more closely the behaviour of this approach when new data are labeled and added to the training sets. In addition to tracking the evolution of the standard and precise accuracy (computed as in Sect. 5.3), we also train, after each iteration of the co-training procedures, a Naive Credal Classifier (with $s = 2$) that can produce set-valued predictions. We then compute values commonly investigated in the assessment of credal approaches [9]: the *single accuracy* which is the percentage of good classification when the decision is determinate; the *set accuracy* which is the percentage of indeterminate predictions that contain the true class; the *determinacy* which is the percentage of confident data (i.e. for which the model decision is determinate).

Figures 2 shows the average curves of the various terms described above according to the number of training iterations, for all the UCI datasets we experiment in this paper. It should be recalled that, in our experiments, we add one instance per iteration in the training set of each model and that this instance is labeled by the other model. To have smoother curves, we compute the terms every 5 iterations. We illustrate the determinacy, the single accuracy and the set accuracy only for the maximality decision rule since we get similar curves with the interval dominance.

A first thing we can notice is that the precise accuracy (whatever the decision rule), compared to the standard accuracy, increases in a steadier and steeper way. This, again, indicates that credal co-training can be more robust than its

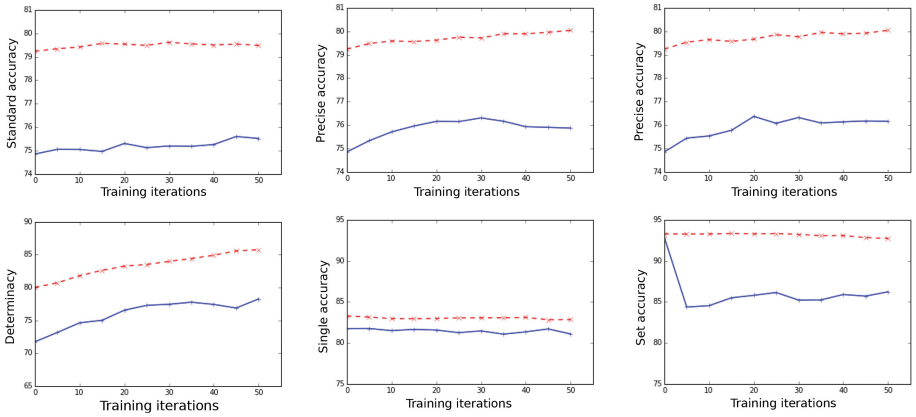

Fig. 2. Average performances on all data sets. Full blue and red dotted lines correspond to the first and second model, respectively. First line (from left to right): NBC accuracy with standard co-training, with credal co-training using interval dominance (middle) and maximality (on the right). Second line: determinacy, single accuracy and set accuracy of the NCC resulting from credal co-training with maximality decision rule. (Color figure online)

standard counterpart. The fact that the single-accuracy is high confirms that the set $S_{H_i \to H_j}$ will contain informative data and that the data for which the credal models give unique predictions during the co-training process are very reliable. Similarly, the high set accuracy suggest that the indeterminate predictions generally contain the true labels.

In addition, the increase of determinacy shows that each classifier becomes more confident as labelled data accumulates, yet it remains cautious on some instances. The fact that the determinacy tends to stabilize after a while suggests that the proposed approach is mainly interesting when starting the process, that is when the information is minimal. This confirms our intuition that one of the main interest of the credal approach is to avoid possible prior bias.

6 Conclusion

In this paper, we propose an extension of the standard co-training process to credal models. Combining the co-training process with the imprecise probability framework enables to define new strategies for selecting informative instances in a pool of unlabeled data. The idea of these strategies is to use the ability of credal models to produce unique predictions only when having enough information, and set-valued predictions when being too uncertain.

We experiment on several UCI datasets the various selection strategies we propose and we compare the credal co-training with the standard co-training process and the supervised framework. Experiments confirm that the co-training

with credal models is generally more efficient and more reliable than the standard co-training framework and the supervised case.

Acknowledgments. This work is founded by the European Union and the French region Picardie. Europe acts in Picardie with the *European Regional Development Fund (ERDF).*

References

1. Amini, M., Usunier, N.: Learning with Partially Labeled and Interdependent Data. Springer, Switzerland (2015)
2. Antonucci, A., Corani, G., Gabaglio, S.: Active learning by the naive credal classifier. In: Sixth European Workshop on Probabilistic Graphical Models (PGM 2012), pp. 3–10 (2012)
3. Augustin, T., Coolen, F.P., de Cooman, G., Troffaes, M.C.: Introduction to Imprecise Probabilities. Wiley, Chichester (2014)
4. Balcan, M.F., Blum, A., Yang, K.: Co-training and expansion: towards bridging theory and practice (2004)
5. Bernard, J.M.: An introduction to the imprecise Dirichlet model for multinomial data. Int. J. Approx. Reason. **39**(2), 123–150 (2005)
6. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: Proceedings of the Eleventh Annual Conference on Computational Learning Theory, COLT 1998, pp. 92–100. ACM (1998)
7. Chapelle, O., Schlkopf, B., Zien, A.: Semi-supervised Learning. MIT Press, Cambridge (2006)
8. Corani, G., Zaffalon, M.: Credal model averaging: an extension of Bayesian model averaging to imprecise probabilities. In: Daelemans, W., Goethals, B., Morik, K. (eds.) ECML PKDD 2008, Part I. LNCS (LNAI), vol. 5211, pp. 257–271. Springer, Heidelberg (2008)
9. Corani, G., Zaffalon, M.: Learning reliable classifiers from small or incomplete data sets: the naive credal classifier 2. J. Mach. Learn. Res. **9**, 581–621 (2008)
10. Corani, G., Zaffalon, M.: Naive credal classifier 2: an extension of naive bayes for delivering robust classifications. DMIN **8**, 84–90 (2008). CSREA Press
11. Dasgupta, S., Littman, M.L., McAllester, D.A.: PAC generalization bounds for co-training. In: Dietterich, T., Becker, S., Ghahramani, Z. (eds.) Advances in Neural Information Processing Systems, vol. 14, pp. 375–382. MIT Press, Cambridge (2002)
12. Destercke, S., Quost, B.: Combining binary classifiers with imprecise probabilities. In: Tang, Y., Huynh, V.-N., Lawry, J. (eds.) IUKM 2011. LNCS, vol. 7027, pp. 219–230. Springer, Heidelberg (2011)
13. Grandvalet, Y., Bengio, Y.: Semi-supervised learning by entropy minimization. Network **17**(5), 529–536 (2005)
14. Kingma, D.P., Rezende, D.J., Mohamed, S., Welling, M.: Semi-supervised learning with deep generative models. CoRR abs/1406.5298 (2014)
15. Liu, A., Reyzin, L., Ziebart, B.D.: Shift-pessimistic active learning using robust bias-aware prediction, pp. 1–7 (2015)
16. Nigam, K., McCallum, A., Thrun, S., Mitchell, T.M.: Text classification from labeled and unlabeled documents using EM. Mach. Learn. **39**(2/3), 103–134 (2000)

17. Qi, R.H., Yang, D.L., Li, H.F.: A two-stage semi-supervised weighted naive credal classification model. Innov. Comput. Inf. Control J. **5**(2), 503–508 (2011)
18. Settles, B.: Active Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan and Claypool Publishers, San Rafael (2012)
19. Soullard, Y., Saveski, M., Artieres, T.: Joint semi-supervised learning of hidden conditional random fields and hidden Markov models. Pattern Recogn. Lett. (PRL) **37**, 161–171 (2013)
20. Troffaes, M.C.: Decision making under uncertainty using imprecise probabilities. Int. J. Approx. Reason. **45**(1), 17–29 (2007)
21. Utkin, L.V.: The imprecise Dirichlet model as a basis for a new boosting classification algorithm. Neurocomputing **151**, 1374–1383 (2015)
22. Wang, W., Zhou, Z.-H.: Analyzing co-training style algorithms. In: Kok, J.N., Koronacki, J., Lopez de Mantaras, R., Matwin, S., Mladenič, D., Skowron, A. (eds.) ECML 2007. LNCS (LNAI), vol. 4701, pp. 454–465. Springer, Heidelberg (2007)
23. Wang, W., Zhou, Z.H.: A new analysis of co-training. In: Proceedings of the 27th International Conference on Machine Learning (ICML 2010), pp. 1135–1142 (2010)
24. Wang, W., Zhou, Z.H.: Co-training with insufficient views. In: Asian Conference on Machine Learning, pp. 467–482 (2013)
25. Zaffalon, M.: The naive credal classifier. J. Stat. Plan. Inference **105**(1), 5–21 (2002). Imprecise Probability Models and their Applications
26. Zhang, M.L., Zhou, Z.H.: Cotrade: confident co-training with data editing. IEEE Trans. Syst. Man Cybern. Part B (Cybern.) **41**(6), 1612–1626 (2011)
27. Zhou, Z.H.: When semi-supervised learning meets ensemble learning. Front. Electr. Electron. Eng. China **6**(1), 6–16 (2011)
28. Zhu, X., Goldberg, A.B., Brachman, R., Dietterich, T.: Introduction to Semi-Supervised Learning. Morgan and Claypool Publishers, San Francisco (2009)