Active Learning by the Naive Credal Classifier

Alessandro Antonucci
IDSIA, Switzerland
alessandro@idsia.ch

Giorgio Corani
IDSIA, Switzerland
giorgio@idsia.ch

Sandra Gabaglio
SUPSI, Switzerland
sandra.gabaglio@supsi.ch

Abstract

In standard classification a training set of supervised instances is given. In a more general setup, some supervised instances are available, while further ones should be chosen from an unsupervised set and then annotated. As the annotation step is costly, *active learning* algorithms are used to select which instances to annotate to maximally increase the classification performance while annotating only a limited number of them. Several active learning algorithms are based on the naive Bayes classifier. We work instead with the naive *credal* classifier, namely an extension of naive Bayes to imprecise probability. We propose two novel methods for active learning based on the naive credal classifier. Empirical comparisons show performance comparable or slightly superior to that of approaches solely based on the naive Bayes.

1 Introduction

In standard classification problems the goal is to learn a classifier able to assign class labels to the unsupervised instances of a *test set*, given a *training set* of supervised instances. If we assume the class variable directly unobservable, the training should be regarded as the output of an annotation process over a set of unsupervised instances.

Active learning (AL) (Settles, 2000) is the process of selecting the instances to be annotated, among all the unsupervised ones. AL algorithms rank the unsupervised instances by means of a *score*; the instances with highest score are then annotated. The classification accuracy increases with the amount of supervised instances, because the variance component of the classification error decreases with the size of the training set. The goal of AL is thus to select the most meaningful instances to be annotated, in order to maximally increase the classifier performance. Several works on active learning have used naive Bayes (NBC) as a classifier; see for instance (Chai et al., 2004; McCallum and Nigam, 1998).

NBC has been extended to cope with sets of probabilities by the so-called *naive credal classifier* (NCC, Corani and Zaffalon (2008)). Instead of a single Dirichlet prior, the parameters are estimated on the basis of a set of priors modeling a condition of near-ignorance *a priori* to achieve more robust and reliable estimates. NCC automatically detects *prior-dependent* instances, namely instances whose most probable class changes when different priors are considered. On these instances, the set of posterior distributions for the class given the attributes is particularly uncertain (i.e., large) and NCC returns multiple classes.

We present two measures of uncertainty based on NCC, which estimate how strong are the dominances among the classes (see Sect. 5.1) and how large the set of posterior distributions for the class is (see Sect. 5.2). In Sect. 7 we compare the AL results obtained with these new methods with their Bayesian counterparts. Moreover, following the ideas in (McCallum and Nigam, 1998), in Sect. 6 we also implement a density-weighted approach; density-weighted approaches are designed to avoid annotating instances which are controversial but very rare, and thus have little impact on the average accuracy of the classifier. We thus also present a density-weighted approach based on NCC.

2 Naive Bayes and Credal Classifiers

We consider a classification task with class variable C, taking values in a finite set $\mathcal{C}$, based on a set of *attributes* $\mathbf{A} := (A_1, \ldots, A_k)$, where for each $i = 1, \ldots, k$, A_i takes values in the finite set $\mathcal{A}_i$. We denote by $|\mathcal{C}|$ the cardinality of $\mathcal{C}$, i.e., the number of possible classes, and similarly for the attributes. A training set of joint observations for these variables is available, i.e., $\mathcal{D} := \{(c^{(i)}, \mathbf{a}^{(i)})\}_{i=1}^{d}$. Learning a *classifier* from data $\mathcal{D}$ means to implement a map $\times_{i=1}^{k}\mathcal{A}_i \to \mathcal{C}$ assigning a class label to any joint observation of the attributes. In particular, probabilistic classifiers are obtained by learning from $\mathcal{D}$ a joint distribution $P_{\mathcal{D}}(C, \mathbf{A})$. Given this distribution, the class label assigned to a (joint) test instance of the attributes, say $\mathbf{a} \in \times_{i=1}^{k}\mathcal{A}_i$ is:[1]

$$c^* := \arg\max_{c \in \mathcal{C}} P_{\mathcal{D}}(c, \mathbf{a}). \tag{1}$$

The *naive Bayes classifier* (NBC) is a probabilistic classifier based on the assumption of conditional independence between the attributes given the class variables, this inducing the factorization $P(c, \mathbf{a}) = P(c)\prod_{i=1}^{k} P(a_i|c)$, where a_i is the value of A_i consistent with $\mathbf{a}$.

NBC, whose parameters are estimated through a standard Bayesian approach, has been extended to imprecise probability by the so-called *naive credal classifier* (NCC, Corani and Zaffalon (2008)). While NBC learns from data a joint distribution $P_{\mathcal{D}}$, NCC learns a joint *credal set*, i.e., a set of joint distributions $\mathcal{P}_{\mathcal{D}}(C, \mathbf{A}) := \{P_\theta(C, \mathbf{A})\}_{\theta \in \Theta}$, where θ is a parameter indexing a family of NBC specifications. NCC does not simply return the class which is the most probable a posteriori as in (1), since multiple posterior distributions now characterize the model; it instead returns the *non-dominated* classes. A class c' *dominates* an alternative class c'' if c' is more probable than c'' for each distribution in the joint credal set; more formally, dominance holds iff:

$$\gamma_{\mathbf{a}}(c', c'') := \inf_{\theta \in \Theta} \frac{P_\Theta(c', \mathbf{a})}{P_\Theta(c'', \mathbf{a})} > 1. \tag{2}$$

Such a dominance test can be efficiently evaluated by the algorithm described in Corani and Zaffalon (2008). NCC compares the classes in a pairwise fashion to identify the set of non-dominated classes $\mathcal{C}^* \subseteq \mathcal{C}$. Corani and Zaffalon (2008) have shown by extensive experiments that instances for which $\mathcal{C}^*$ contains more than a single class are often misclassified by NBC.

3 Active Learning (AL)

Let us focus on the way the training set can be obtained. Assuming the class directly unobservable, $\mathcal{D}$ should be regarded as the result of the *annotation*[2] of a set of unsupervised observations of the attributes. In general situations, only some of the instances might be annotated, this requiring the choice of which instances to annotate. An *active learning* (AL) algorithm provides a strategy for identifying the instances to be annotated. In particular, we consider the situation where a supervised training set is already available and l instances should be picked from an unsupervised set, called the *active learning set*. The performance indicator of an AL algorithm is the growth of the accuracy of the classifier when the l instances are annotated

[1] In both (1) and (2) the joint probability can replace the corresponding conditional for the class because of the proportionality relation among them.

[2] We call annotation the process of assigning the proper class label to an unsupervised instance of the attributes. This is achieved by an *annotation oracle* corresponding to a perfect (i.e., 100% accuracy) classifier.

and added to the training set. Even a random picking of the instances generally increases the accuracy; thus an AL algorithm should produce a quicker increase of accuracy over selecting the instances in a random fashion.

4 Active Learning with Naive Bayes

Let us denote by $\mathcal{D}_A$ the *active learning set*, namely the set of instances from which to select the instances to be annotated. We consider a setting in which at each iteration the AL algorithm selects from $\mathcal{D}_A$ the subset of l instances with the highest *score*. In the following we describe two well-known AL algorithms, which can be used with NBC.

4.1 Uncertainty Sampling (US)

One of the simplest and most commonly used query framework is uncertainty sampling: the active learner queries the instances whose labels are the most uncertain (Settles, 2000, pag. 12). The score is defined as follows:

$$\text{score}(\mathbf{a}) := -P(c^*|\mathbf{a}), \tag{3}$$

where c^* is the most probable class as in (1). Because of the minus, the most uncertain instances will have the highest score. For binary classification, uncertainty sampling queries the instance whose posterior probability of being positive is nearest to 0.5.

4.2 Query by Committee (QbC)

Following the definition of (Settles, 2000, pag. 15), "*the QbC approach involves maintaining a committee of models which are all trained on the current labeled set, but represent competing hypotheses. Each committee member is then allowed to vote on the labelings of query candidates. The most informative query is considered to be the instance about which they most disagree.*" We implement QbC by generating q different bootstrap replicates $\{\mathcal{D}_j\}_{j=1}^{q}$ of the training set and then learning a different NBC from each of them. Denote by $P^{(j)}(C|\mathbf{a})$ the posterior distributions computed by the j-th NBC; the center of mass of such posteriors is:

$$\tilde{P}_{\mathbf{a}}(c) := \frac{1}{q}\sum_{j=1}^{q} P^{(j)}(c|\mathbf{a}). \tag{4}$$

The score proposed in (McCallum and Nigam, 1998) is the average of the KL divergence between the members of the committee and the center of mass, i.e.,[3]

$$\text{score}(\mathbf{a}) := \frac{1}{q}\sum_{j=1}^{q} \text{KL}[P^{(j)}(C|\mathbf{a}), \tilde{P}_{\mathbf{a}}(C)]. \tag{5}$$

The more the posterior distributions of the committee members disagree, the higher the score.

5 NCC-based Active Learning

As pointed out in Sect. 2, NCC natively identifies prior-dependent instances, for which it returns multiple class labels in output. The more the returned classes, the harder the instance to be classified, as shown in Corani and Zaffalon (2008). However the number of non-dominated classes is not a viable score for active learning: such an approach would not be able to discriminate among instances which have received the same number of class labels. Further refinements are thus necessary for a NCC-based AL approach; we present two alternative approaches in the following.

5.1 Credal Uncertainty Sampling

Let us start by considering a binary class, i.e., $\mathcal{C} = \{c', c''\}$. The instances for which the dominance is more clear are characterized by higher values of the maximum ratio between the posterior probabilities of the two classes, introduced in (2). We thus propose the following AL score:

$$\text{score}(\mathbf{a}) := -\max[\gamma_{\mathbf{a}}(c', c''), \gamma_{\mathbf{a}}(c'', c')]. \tag{6}$$

where $\gamma_{\mathbf{a}}(c', c'')$ has been defined in (2). The more negative this score, the clearer the superiority of a class over another one; consistently with the previous ones, higher scores correspond to instances deemed to be more uncertain.

[3] The Kullback-Leibler divergence with discrete variables is $\text{KL}[P', P''] := -\sum_{c\in\mathcal{C}} P'(c) \cdot \log \frac{P''(c)}{P'(c)}$.

In particular with scores smaller than -1, one of the two classes dominates the other as in (2). If the score is greater than -1, there is no dominance among the two classes. Thus, any instance for which NCC has returned two classes should be regarded as more uncertain than any instance for which NCC has returned a single class. For data sets with more than two classes, the score is generalized as follows:

$$-\sum_{c'\in\mathcal{C}}\sum_{c''\in\mathcal{C}\setminus\{c'\}}\max[\gamma_{\mathbf{a}}(c',c''),\gamma_{\mathbf{a}}(c'',c')]. \quad (7)$$

We call this approach *credal uncertainty sampling* (CUS); it tries to identify how strong is the dominance among the classes, and thus it can be seen as a credal counterpart of the uncertainty-sampling based on NBC.

5.2 Credal Query by Committee

QbC artificially generates multiple NBCs through a bootstrap resampling of the dataset. Notice that NCC constitutes a committee of NBCs as well, although different from QbC. Each committee member of NCC is in fact induced by the updating of a different prior (those in the *imprecise Dirichlet model*, see App. A); therefore, each member computes a different posterior. Yet, the set $\mathcal{P}(C|\mathbf{a}) := \{P_\theta(C|\mathbf{a})\}_{\theta\in\Theta}$ has an infinite number of elements, this preventing a straightforward application of the ideas in Sect. 4.2. Nevertheless, the set of posteriors is convex and its *extreme* points, namely those which cannot be obtained as a convex combination of other posteriors, are only a finite number.[4] We call *Credal Query by Committee* (CQbC) a QbC-like approach which adopts as committee members only the extremes of the NCC set of posterior distribution.

To support the idea of removing non-extreme members from the committee, it could be worth noticing that computing the bounds of an expectation with respect to a set of distributions is a LP task, whose optimum lies on an extreme point of the feasible region.

To obtain the extremes of $\mathcal{P}(C|\mathbf{a})$, we first evaluate its lower and upper bounds, i.e.,

$$\underline{P}(c|\mathbf{a}) := \inf_{\theta\in\Theta} P_\theta(c|\mathbf{a}), \quad (8)$$

$$\overline{P}(c|\mathbf{a}) := \sup_{\theta\in\Theta} P_\theta(c|\mathbf{a}). \quad (9)$$

The values and the formulae to compute these bounds are in App. A. Then we consider the set of distributions consistent with these bounds, i.e., set $\mathcal{P}'(C|\mathbf{a})$ defined as:

$$\left\{P(C)\;\middle|\;\begin{array}{c}\underline{P}(c|\mathbf{a})\le p(c)\le\overline{P}(c|\mathbf{a})\forall c\in\mathcal{C}\\ \sum_{c\in\mathcal{C}}P(c)=1\end{array}\right\}. \quad (10)$$

The extremes of $\mathcal{P}'(C|a)$ can be obtained from the constraints in (10) by the fast algorithm in (de Campos et al., 1994), their number being bounded by the factorial of $|\mathcal{C}|$.

Although, in general, $\mathcal{P}'(C|\mathbf{a})$ is only an outer approximation of $\mathcal{P}(C|\mathbf{a})$, the two credal sets are known to be very close (e.g., see the discussion in Antonucci and Cuzzolin (2010)). We can therefore evaluate the score in (5) with the elements $\{P^{(j)}(C|\mathbf{a})\}_{j=1}^q$ coinciding with the extremes of $\mathcal{P}'(C|\mathbf{a})$. This QbC-like approach will be called *credal query by committee* (CQbC).

In Sect. 7 we show that also in the traditional QbC what matters are in fact the extreme members, namely those whose opinion cannot be obtained as a convex combination of other members: removing members of the committee which are in the convex hull of the others basically does not affect the QbC ranks.

6 Density-weighted approaches

For a further improvement in the AL performance, (McCallum and Nigam, 1998) proposed to weight the score of each instance on the basis of its representativeness. In the words of (Settles, 2000, pag. 25) "*the main idea is that informative instances should not only be those which are uncertain, but also those which are representative of the underlying distribution (i.e., inhabit dense regions of the input space)*". In particular a similarity measure among the attributes instances is defined and then summed over all the instances. As an alternative to this information-theoretic approach, here we propose a purely probabilistic approach, where the

[4] See the *lower envelope theorem* in (Walley, 1991).

level of representativeness of each instance corresponds to the marginal probability of the joint observation of the instances, i.e., we rescale the NBC-based scores as follow:[5]

$$\text{score}'(\mathbf{a}) = \text{score}(\mathbf{a}) \cdot P(\mathbf{a}). \tag{11}$$

For the NBC, the weights are simply:

$$P(\mathbf{a}) = \sum_{c \in \mathcal{C}} \left[P(c) \prod_{i=1}^{k} P(a_i|c) \right]. \tag{12}$$

This idea can be easily extended to the NCC by simply replacing in (12) the probability of the joint observation of the attributes with the corresponding *lower* probability, i.e.,

$$\underline{P}(\mathbf{a}) = \inf_{\theta \in \Theta} P_\theta(\mathbf{a}). \tag{13}$$

A formula for this bound is in App. B.

7 Experiments

We empirically compare the AL algorithms based on NCC with their counterparts based on NBC on different data sets from the UCI repository. Notice that we use the AL algorithms based on NCC to rank the instances and to select which ones should be annotated; then, we use the annotated instances to update the parameters of a standard NBC classifier.

For instance, to compare uncertainty sampling (US) and credal uncertainty sampling (CUS), we proceed as follows: we randomly draw a training set of n_0=10 instances and a test set of 100 instances; we generate both training and test set in a stratified way, namely they contain the same proportion of instances from the various classes as the original data set. We then estimate the parameters of NBC from the training set; this starting classifier is denoted as $\text{NBC}^{(0)}$. Then, we rank the unsupervised instances in $\mathcal{A}$ according to *uncertainty sampling*, using $\text{NBC}^{(0)}$; we then annotate the l=5 instances with highest score and use them to revise $\text{NBC}^{(0)}$, thus obtaining $\text{NBC}^{(1)}_{\text{US}}$, which denotes a NBC actively learned by uncertainty sampling at the first iteration. We then update the active learning set as $\mathcal{D}^{(1)}_{\text{US}} := \mathcal{D}_A \setminus \overline{\mathcal{D}}^{(1)}_{\text{US}}$, where $\overline{\mathcal{D}}^{(1)}_{\text{US}}$ denotes the l instances selected by uncertainty sampling for the first update; we then evaluate the accuracy of $\text{NBC}^{(1)}_{\text{US}}$ on the test set. This procedure (scoring instances, updating classifier and active learning set, assessing accuracy on the test set) is iterated until the active learning set is empty.

To assess *credal uncertainty sampling* we follow the very same procedure, beginning from the same starting point, namely $\text{NBC}^{(0)}$, with the only difference of using the score in (6) to rank the instances to be annotated. Namely, we use the l=5 instances with highest CUS score to update the NBC parameters, yielding $\text{NBC}^{(1)}_{\text{CUS}}$, which denotes a NBC actively learned by *credal* uncertainty sampling, after the first active learning iteration. We then update the active learning set as $\mathcal{D}^{(1)}_{\text{CUS}} = \mathcal{D}_A \setminus \overline{\mathcal{D}}^{(1)}_{\text{CUS}}$. The accuracy of $\text{NBC}^{(1)}_{\text{CUS}}$ is then assessed on the test set, and so on. We remark that the starting training set and classifier $\text{NBC}^{(0)}$ is identical for the different AL algorithms, thus allowing to fairly compare them. We repeat the procedure 50 times for each data set and we report the average results.

We compare the following pairs of methods: US versus CUS; QbC versus CQbC. To choose the number of member of the QbC committee, we first noticed that: "*there is no general agreement in the literature on the appropriate committee size to use, which may in fact vary by model class or application. However, even small committee sizes (e.g., two or three) have been shown to work well in practice*" (Settles, 2000, pag. 16). After some preliminary experiments, q=7 seemed a viable choice for this parameter.

It is hard to summarize in a single indicator the performance of an AL algorithm; it is instead much more meaningful showing the whole trajectory of accuracy for training sets of growing dimensions, as shown in Fig. 1. Although we analyzed 16 data sets, for the lack of space we graphically present only part of the results, selecting some representative examples.

[5] We assume that the scores are nonnegative values. If this is not the case we can always sum to all of them the minimum score to obtain nonnegative values.

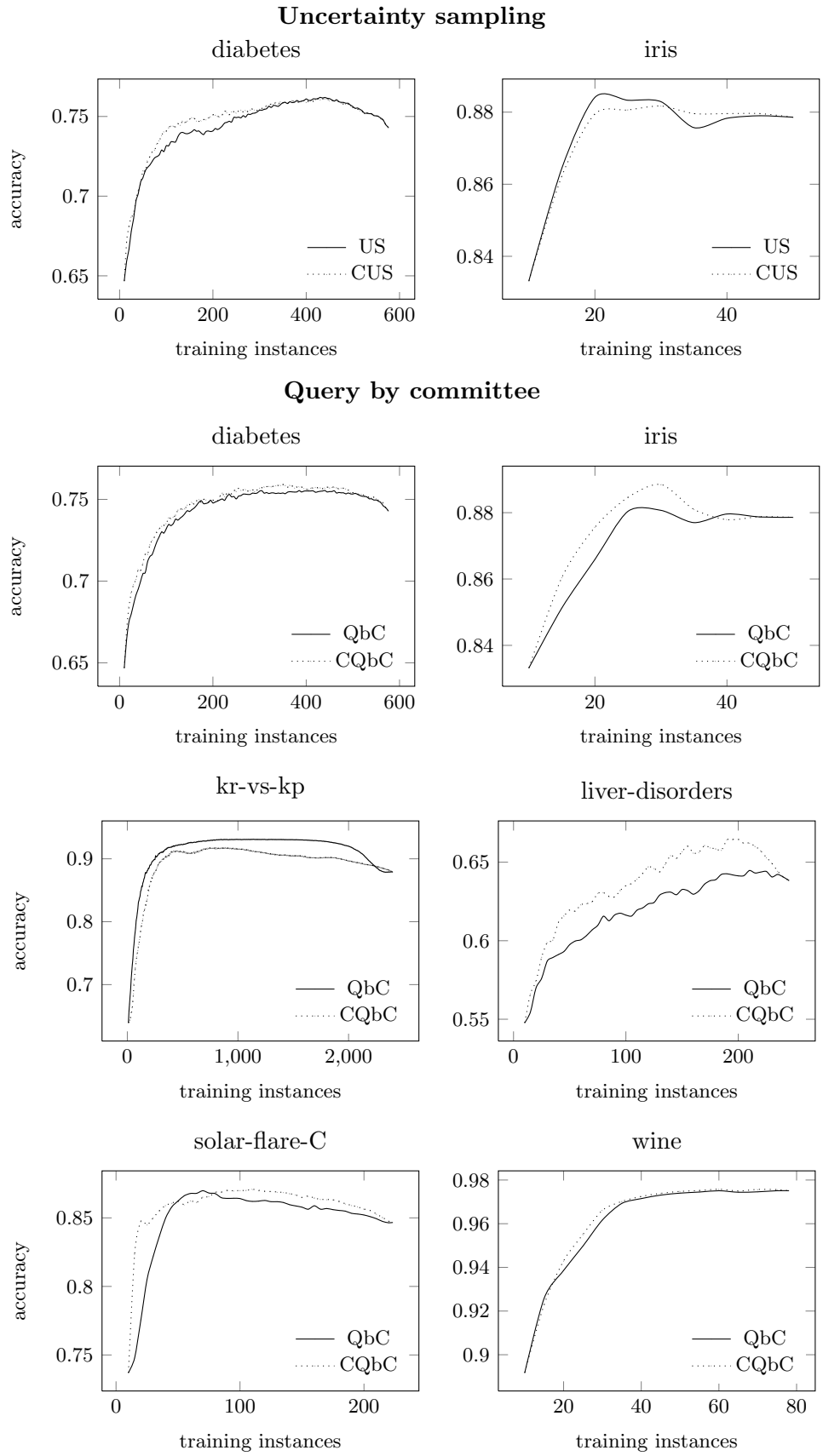

Figure 1: Experimental results on different data sets.

The comparison between US and CUS did not show a clear winner among the two approaches. Most commonly, the accuracy trajectories of the two methods cross different times in a single graph; we conclude that the two approaches perform in a substantially equivalent way. Some examples are shown in the top row of Fig. 1.

Notably, when using criteria different from the random sampling, the training set is not necessarily representative of the underlying population: the variance component of the classification error might therefore not decrease when the training set size increases. This explains the local decreases in some trajectories. On the contrary, the accuracy trajectories for the random sampling are monotonically increasing, but being always under the trajectories associated to both US and QbC (Settles, 2000) are not shown in the plots.

The comparison between QbC and CQbC shows instead a certain superiority for CQbC, as can be inferred from Fig. 1; an exception is however found on the kr-vs-kp data set, whose results are shown in the third row of Fig. 1. Overall the results suggest that the committee obtained by updating a convex set of priors outperforms the committee obtained by building a finite number of bootstrap replicates. A possible explanation is that the former approach has an infinite number of members, which might be beneficial for the accuracy of the committee.

An interesting empirical finding is that however in the committee member what really matters are the extreme members. The similarity among two ranks can be assessed by the Spearman correlation, whose maximum value is 1. We found a Spearman correlation consistently higher than 0.9 between the rank computed, on the instances of the active learning set, by QbC and QbC restricted to its extreme members. In binary classification, the two extreme members of the committee are those which assign the highest and the lowest probability to the positive class; the extreme members change instance by instance, and therefore this finding does not allow to prune the committee. Yet, it provides some empirical support for the CQbC approach, which relies on the upper and lower probability for the class.

Finally, the probabilistic version of the density-weighted approach proposed in Sect. 6 did not generally provide a significant improvement in the performance. Our explanation for this result is that the computed probability of an instance are negatively affected by the adopted naive architecture, which is known to be a bad estimator of the joint probability. Since weighting the scores by a density measure is generally recognized to boost the AL performance, we see as a future research direction the adoption of more realistic topologies for the computation of the probability of the instance.

8 Conclusions

In this paper we proposed two new active learning algorithms based on the naive credal classifier, rather than on the traditional naive Bayes. Results are especially encouraging for the credal query-by-committee approach; future research direction include the refinement of the density-weighted approach, in order to further boost the active learning performance.

Acknowledgements

The research in this paper has been partially supported by the Swiss NSF grants no. 200020-132252 and by the Hasler foundation grant n. 10030. We also thank the first reviewer for his/her constructive comments.

References

A. Antonucci and F. Cuzzolin. 2010. Credal sets approximation by lower probabilities: application to credal networks. In *Proceedings of IPMU 2010*, pages 716–725.

X. Chai, L. Deng, Q. Yang, and C.X. Ling. 2004. Test-cost sensitive naive bayes classification. In *Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on*, pages 51–58. IEEE.

G. Corani and M. Zaffalon. 2008. Learning reliable classifiers from small or incomplete data sets: The naive credal classifier 2. *The Journal of Machine Learning Research*, 9:581–621.

L.M. de Campos, J.F. Huete, and S. Moral. 1994. Probability intervals: a tool for uncertain reason-

ing. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 2:167–196.

A. McCallum and K. Nigam. 1998. Employing em and pool-based active learning for text classification. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 350–358. Morgan Kaufmann Publishers Inc.

B. Settles. 2000. Active learning literature survey. *Computer Sciences Technical Report*, 1648.

P. Walley. 1991. *Statistical Reasoning with Imprecise Probabilities*. Chapman and Hall, New York.

Appendix A: Lower conditional

Consider the computation of the posterior probability for the class given the attributes. For the NBC we have:

$$P(c|\mathbf{a}) := \frac{P(c,\mathbf{a})}{\sum_{c'\in\mathcal{C}} P(c',\mathbf{a})}, \tag{14}$$

which, by exploiting the NBC factorization, with straightforward algebra, rewrites as:

$$P(c|\mathbf{a}) = \left[1 + \frac{\sum_{c'\neq c} P(c')\prod_{i=1}^{k} P(a_i|c')}{P(c)\prod_{i=1}^{k} P(a_i|c)}\right]^{-1}. \tag{15}$$

The NBC probabilities are obtained from the counts in $\mathcal{D}$ as:

$$P(a_i|c) := \frac{n(a_i,c)+st(a_i,c)}{n(c)+s}, \tag{16}$$

$$P(c) := \frac{n(c)+st(c)}{n(\cdot)+s}, \tag{17}$$

this corresponding to a Bayesian learning approach with Dirichlet priors with parameters $\{t(c), t(a_i,c)\}$. Unlike NBC, NCC consider a set of priors, whose parameters are free to vary in the following sets:[6]

$$\left\{ t(C) \;\middle|\; \begin{array}{l} \frac{\epsilon}{|\mathcal{C}|} \leq t(c) \leq \frac{\epsilon}{|\mathcal{C}|} + (1-\epsilon) \\ \forall c \in \mathcal{C}, \\ \sum_{c\in\mathcal{C}} t(c) = 1 \end{array} \right\}, \tag{18}$$

$$\left\{ t(A_i|c) \;\middle|\; \begin{array}{l} \frac{\epsilon}{|\mathcal{A}_i|} \leq t(a_i,c) \leq \frac{\epsilon}{|\mathcal{A}_i|} + (1-\epsilon) \\ \forall a_i \in \mathcal{A}_i \\ \sum_{a_i\in\mathcal{A}_i} t(a_i,c) = 1 \end{array} \right\}. \tag{19}$$

[6]This is the so-called imprecise Dirichlet model which, after an ϵ regularization to avoid problems with zero counts. In particular, this is called the *local* version of the IDM model.

The problem is therefore minimize (15) with the probabilities defined as in (16) and (17) with respect to the constraints in (18) and (19). By simply exploiting the monotonicity of function $f(x) := \frac{1}{1+x}$ and with simple algebra we obtain:

$$\underline{P}(c|\mathbf{a}) = \sum_{c'\in\mathcal{C}\setminus\{c\}} \frac{n(c')+s\frac{\epsilon}{|\mathcal{C}|}}{n(c)+s[\frac{\epsilon}{|\mathcal{C}|}+(1-\epsilon)]} \cdot \left(\frac{n(c)+s}{n(c')+s}\right)^{k} \prod_{i=1}^{k} \frac{n(a_i,c')+s\frac{\epsilon}{|\mathcal{A}_i|}}{n(a_i,c)+s[\frac{\epsilon}{|\mathcal{A}_i|}+(1-\epsilon)]}. \tag{20}$$

Similar considerations hold for the maximum:

$$\overline{P}(c|\mathbf{a}) = \sum_{c'\in\mathcal{C}\setminus\{c\}} \frac{n(c')+s\tilde{\delta}_{c'\tilde{c}}}{n(c)+\frac{\epsilon}{|\mathcal{C}|}} \left(\frac{n(c)+s}{n(c')+s}\right)^{k} \cdot \prod_{i=1}^{k} \frac{n(a_i,c')+s[\frac{\epsilon}{|\mathcal{A}_i|}+1-\epsilon]}{n(a_i,c)+s\frac{\epsilon}{|\mathcal{A}_i|}}, \tag{21}$$

with

$$\tilde{\delta}_{c'\tilde{c}} := \begin{cases} \frac{\epsilon}{|\mathcal{C}|}+(1-\epsilon) & \text{if} \quad c'=\tilde{c} \\ \frac{\epsilon}{|\mathcal{C}|} & \text{otherwise} \end{cases}.$$

Appendix B: Lower marginal

Consider the evaluation of the lower probability for the joint instance of the attributes defined as in (13). This corresponds to evaluate the minimum of the expression in (12), again with the probabilities defined as in (16) and (17) with respect to the constraints in (18) and (19).

When minimizing this objective function, we first consider the (trivial) minimization with respect to the constraints in (19). Then, we obtain the following linear program:

$$\text{minimize:} \qquad \sum_{c\in\mathcal{C}} \left[\prod_{i=1}^{n} \frac{n(a_i,c)+s\frac{\epsilon}{|\mathcal{A}_i|}}{n(c)+s}\right] t_c \tag{22}$$

$$\text{w.r.t.:} \qquad \begin{array}{c} \frac{\epsilon}{|\mathcal{C}|} \leq t_c \leq \frac{\epsilon}{|\mathcal{C}|} + (1-\epsilon) \\ \sum_{c\in\mathcal{C}} t_c = 1, \end{array} \tag{23}$$

whose solution $\{t_c^*\}$ can be plugged in (12) to obtain (13).