Credal Classification based on AODE and compression coefficients

G. Corani[a,∗], A. Antonucci[a]

[a]*IDSIA*
Istituto Dalle Molle di Studi sull'Intelligenza Artificiale
CH-6928 Manno (Lugano), Switzerland

Abstract

Bayesian model averaging (BMA) is a common approach to average over alternative models; yet, it usually gets excessively concentrated around the single most probable model, therefore achieving only sub-optimal classification performance. The compression-based approach (Boullé, 2007) overcomes this problem; it averages over the different models by applying a logarithmic smoothing over the models' posterior probabilities. This approach has shown excellent performances when applied to ensembles of naive Bayes classifiers. AODE is another ensemble of models with high performance (Webb et al., 2005): it consists of a collection of non-naive classifiers (called SPODE) whose probabilistic predictions are aggregated by simple arithmetic mean. Aggregating the SPODEs via BMA rather than by arithmetic mean deteriorates the performance; instead, we propose to aggregate the SPODEs via the compression coefficients and we show that the resulting classifier obtains a slight but consistent improvement over AODE. However, an important issue in any Bayesian ensemble of models is the arbitrariness in the choice of the prior over the models. We address this problem by adopting the paradigm of *credal* classification, namely by substituting the unique prior with a set of priors. Credal classifier are able to automatically recognize the *prior-dependent* instances, namely the instances whose most probable class varies, when different priors are considered; in these cases, credal classifiers remain reliable by returning a set of classes rather than a single class. We thus develop the credal version

---

∗Corresponding author: giorgio@idsia.ch

of both the BMA-based and the compression-based ensemble of SPODEs, substituting the single prior over the models by a set of priors. By experiments we show that both credal classifiers provide overall higher classification reliability than their determinate counterparts. Moreover, the compression-based credal classifier compares favorably to previous credal classifiers.

1. Introduction

Bayesian model averaging (BMA) (Hoeting et al., 1999) is a sound solution to the uncertainty which characterizes the identification of the supposedly best model for a certain data set; given a set of alternative models, BMA weights the inferences produced by the various models, using the models' posterior probabilities as weights. BMA assumes the data to be generated by one of the considered models; under this assumption, it provides better predictive accuracy than any single model (Hoeting et al., 1999). However, such an assumption is generally not true; for this reason, on real data sets BMA does not generally perform very well; see the discussion and the references in Cerquides et al. (2005) for more details. The problem is that BMA gets excessively concentrated around the single most probable model (Domingos, 2000; Minka, 2002): especially on large data sets, "*averaging using the posterior probabilities to weight the models is almost the same as selecting the MAP model*" (Boullé, 2007). To overcome the problem of BMA getting excessively concentrated around the most probable model, a *compression-based* approach has been introduced in (Boullé, 2007); it computes more evenly-distributed weights, by applying a logarithmic smoothing to the models posterior probabilities. The compression-based weights, which can be justified from an information-theoretic viewpoint, have been used in Boullé (2007) to average over different naive Bayes classifiers, characterized by different feature sets, obtaining excellent rank in international competitions on classification.

Another ensemble of Bayesian networks classifiers known for its good performance is AODE (Webb et al., 2005), which is instead based on a set of SPODE (SuperParent-

One-Dependence Estimator) models. Each SPODE adopts a certain feature as a *super-parent*, namely it models all the remaining features as depending on both the class *and* the super-parent. AODE then simply averages the posterior probabilities computed by the different SPODEs. Alternative methods to aggregate SPODEs, more complex than AODE, have been considered (Yang et al., 2007), but AODE generally outperforms them: "*AODE, which simply linearly combines every SPODE without any selection or weighting, is actually more effective than the majority of rival schemes*". As reported in (Cerquides et al., 2005; Yang et al., 2007), AODE outperforms aggregating SPODEs via BMA; in both (Yang et al., 2007; Cerquides et al., 2005) the best results were instead obtained using an algorithm (called MAPLMG), which estimates the most probable linear *mixture* of SPODEs; this overcomes the problem of assuming a single SPODE to be the true model. In this paper, we address this problem by means of the compression coefficients.

As a preliminary step we develop BMA-AODE, namely BMA over SPODEs, with some computational differences with respect to the framework of Yang et al. (2007) and Cerquides et al. (2005); our results confirm however that BMA over SPODEs is outperformed by AODE. Then we develop the novel COMP-AODE classifier, which weights the SPODEs using the compression-based coefficients, and we show that it yields a slight but consistent improvement in the classification performance over the standard AODE. Considering the high performance of AODE, we regard this result as noteworthy.

An important issue in any Bayesian ensemble is choosing the prior over the models. A common choice is to adopt a uniform mass function, as we do in both BMA-AODE and COMP-AODE; this however can be criticized from different standpoints; see for instance the rejoinder in Hoeting et al. (1999). In Boullé (2007), a prior which favors simpler models over complex ones is adopted. Although all these choices are reasonable, the specification of any single prior implies some arbitrariness, which entails the risk of prior-dependent, and hence potentially fragile, conclusions.

In fact, the specification of the prior over the models is a serious open problem for Bayesian ensembles of models. We address this problem by adopting the paradigm of *credal classification* (Corani et al., 2012; Corani and Zaffalon, 2008b), namely drop-

ping the unique prior in favor of a *set* of priors (prior *credal set*) (Levi, 1980). While a traditional non-informative priors represents a condition of *indifference* between the alternative models, a credal set describes a condition of prior *ignorance*, letting thus vary the prior probability of each model over a wide interval, instead of fixing it to a specific number. Credal classifiers are able to automatically detect the instances whose most probable class varies when different priors are considered; such instances are called *prior-dependent*. Credal classifiers remain reliable on prior-dependent instances by returning a *set* of classes; traditional classifiers have instead typically low accuracy on the prior-dependent instances (Corani and Zaffalon, 2008a,b).

We then develop BMA-AODE* and COMP-AODE*, namely the credal counterparts of respectively BMA-AODE and COMP-AODE. By extensive experiments we show that both credal classifiers are sensible extension of their single-prior counterparts; in fact, they return a small-sized but highly accurate set of classes on the prior-dependent instances, over which instead their single-prior counterparts have reduced accuracy. We conclude by showing that COMP-AODE* compares favorably to both BMA-AODE* and other existing credal classifiers.

2. Methods

We consider a classification problem with k *features*; we denote by C the *class* variable (taking values in $\mathcal{C}$) and by $\mathbf{A} := (A_1, \ldots, A_k)$ the set of features, taking values respectively in $\mathcal{A}_1, \ldots, \mathcal{A}_k$. For a generic variable A, we denote as $P(A)$ the probability mass function over its values and as $P(a)$ the probability that $A = a$. We assume the data to be complete and the training data $\mathcal{D}$ to contain n instances. We learn the model parameters from the training data by adopting Dirichlet priors and setting the equivalent sample size to 1. Under 0-1 loss a traditional probabilistic classifier returns, for a test instance $\tilde{\mathbf{a}} = \{\tilde{a_1}, \ldots, \tilde{a_k}\}$ whose class is unknown, the most probable class c^*:

$$c^* := \arg \max_{c \in \mathcal{C}} P(c|\tilde{\mathbf{a}}).$$

Classifiers based on imprecise-probabilities (*credal* classifiers) change this paradigm, by occasionally returning more classes; this happens in particular when the most proba-

ble class is *prior-dependent*. We discuss this point more in detail later, when presenting credal classifiers.

2.1. From Naive Bayes to AODE

The Naive Bayes classifier assumes the stochastic independence of the features given the class; it therefore factorizes the joint probability as follows:

$$P(c, \mathbf{a}) := P(c) \cdot \prod_{j=1}^{k} P(a_j|c), \qquad (1)$$

corresponding to the topology of Fig.1(a). Despite the biased estimate of probabilities due to the above (so-called *naive*) assumption, naive Bayes performs well under 0-1 loss (Domingos and Pazzani, 1997); it thus constitutes a reasonable choice if the goal is simple classification, without the need for accurate probability estimates; it is especially competitive on data sets of small and medium size , thanks to its low variance error (Friedman, 1997).

To improve the model, weaker assumptions about the conditional independence of the features have to be considered; for instance, the tree-augmented naive classifier (TAN) allows each feature to depend on the class *and* on possibly another single feature, constraining however the subgraph involving only the features to be a tree; an example is shown in Fig.1(b). Generally, TAN outperforms naive Bayes in classification (Friedman et al., 1997).
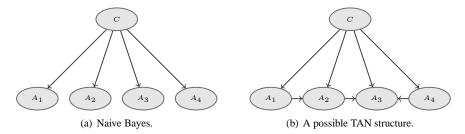

(a) Naive Bayes.　　　　　(b) A possible TAN structure.

Figure 1: Naive Bayes vs TAN.

The AODE classifier (Webb et al., 2005) is an ensemble of k SPODE (SuperParent One Dependence Estimator) classifiers; each SPODE is characterized by a certain *super-parent* feature, so that the other features are modeled as depending on both the class and the super-parent, as shown in in Fig.2. In fact, each single SPODE is a TAN.
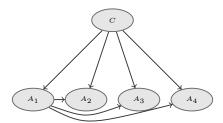
Figure 2: SPODE with super-parent A_1.

We denote the set of SPODEs as $\mathcal{S} := \{s_1, \ldots, s_k\}$, where s_j indicates the SPODE with super-parent A_j. The joint probability of SPODE s_j factorizes as:

$$P(c, \mathbf{a}|s_j) = P(c) \cdot P(a_j|c) \cdot \prod_{l=1..k, l \neq j}^{k} P(a_l|a_j, c).$$

In order to classify the test instance $\tilde{\mathbf{a}}$, AODE averages the posterior probability $P(c|\tilde{\mathbf{a}}, s_j)$ computed by each single SPODE:

$$P(c|\mathbf{a}) = \frac{1}{k} \sum_{j=1}^{j=k} P(c, \mathbf{a}|s_j)$$

In this paper we focus on more sophisticated approaches for aggregating the predictions of the SPODEs.

2.2. Bayesian Model Averaging (BMA) with SPODEs

BMA assumes that one of the models in the ensemble is the true one. Under this assumption, the optimal strategy is to weight the inferences produced by the models of the ensemble using as weights the models' posterior probabilities. By applying BMA on top of different SPODEs, we thus assume one of the SPODEs to be the true model. We thus introduce a variable S over $\mathcal{S}$, where $P(S = s_j)$ denotes the *prior* probability of SPODE s_j to be the true model. Considering that every SPODE has the same number of variables, the same number of arcs and the same in-degree[1], we adopt a *uniform* prior, thus assigning prior probability $1/k$ to each SPODE. In fact, the uniform prior over the models is frequently adopted within BMA. To classify the test

---

[1]The in-degree is the maximum number of parents per node: it is two for any SPODE.

instance $\tilde{\mathbf{a}}$, BMA computes the following posterior mass function:

$$P(c|\tilde{\mathbf{a}}) = \sum_{j=1}^{k} P(c|\tilde{\mathbf{a}}, s_j) \cdot P(s_j|\mathcal{D}) \propto \sum_{j=1}^{k} P(c|\tilde{\mathbf{a}}, s_j) \cdot P(\mathcal{D}|s_j) \cdot P(s_j),$$

where $P(\mathcal{D}|s_j)$ is the *marginal* likelihood of s_j, namely

$$P(\mathcal{D}|s_j) = \int P(\mathcal{D}|s_j, \theta_j) \cdot P(\theta_j|s_j) \cdot d\theta_j,$$

with θ_j denoting the parameters of SPODE s_j. This computational schema has been adopted to implement BMA over SPODEs in (Cerquides et al., 2005; Yang et al., 2007), and has been outperformed by AODE.

The marginal likelihood measures how good the model is at representing the *joint* distribution; yet, a classifier has instead to estimate the posterior probability of the classes *conditionally* on the features. Therefore, a model can perform badly at classification despite having high marginal likelihood (Cowell, 2001; Kontkanen et al., 1999); for this reason, scoring rules more appropriate for classification should be considered. Following Boullé (2007), we thus substitute the marginal likelihood with *conditional* likelihood:

$$L_j := \prod_{i=1}^{n} P(c^{(i)}|\mathbf{a}^{(i)}, s_j, \hat{\theta}_j), \tag{2}$$

where $P(c^{(i)}|\mathbf{a}^{(i)}, s_j, \hat{\theta}_j)$ denotes the probability assigned by model s_j to the true class of the i-th instance, and $\hat{\theta}_j$ is the estimate of the parameters of model s_j.

We call BMA-AODE the classifier which estimates the posterior probabilities of the class, given the test instance $\tilde{\mathbf{a}}$, as follows:

$$P(c|\tilde{\mathbf{a}}) \propto \sum_{j=1}^{k} P(c|\tilde{\mathbf{a}}, s_j) \cdot L_j \cdot P(s_j). \tag{3}$$

Especially on large data sets, the difference between the likelihoods of the different SPODEs might be of several order of magnitudes. We remove from the ensemble the SPODEs whose conditional likelihood is smaller than $L_{\max}/10^4$, where $L_{\max}$ is the maximum conditional likelihood among all SPODEs; discarding models with very low posterior probability is in fact common when dealing with BMA; this procedure can be seen as a *belief revision* (Dubois and Prade, 1997). Given the joint

7

beliefs $P(X, Y)$, the *revision* $P'(X, Y)$ induced by a marginal $P'(Y)$ is defined by $P'(x, y) := P(x|y) \cdot P'(y)$. In other words, if $P'(y)$ is known to be a better model than $P(y)$ for the marginal beliefs about y, this information can be used in the above described way to redefine the joint. Accordingly, in BMA-AODE, the marginal beliefs about S have been replaced by a better candidates, inducing a revision in the corresponding joint model.

2.2.1. Exponentiation of the Log-Likelihoods

Regardless whether the marginal likelihood or the conditional likelihood is considered, it is common to compute the *log-likelihood* rather than the likelihood, in order to avoid numerical problems due to the multiplication of many probabilities. However, if the log-likelihoods are very negative, as it happen on large data sets, their exponentiation can suffer numerical problems too. This issue has been addressed in Yang et al. (2007) by means of high numerical precision: "*BMA often lead to arithmetic overflow when calculating very large exponentials or factorials. One solution is to use the Java class BigDecimal which unfortunately can be very slow.*" Algorithm 1 describes a procedure for exponentiating the log-likelihoods, which is both numerically robust and computationally fast. The procedure has been communicated to us by D. Dash, who published several works on BMA (Dash and Cooper, 2004).

2.3. BMA-AODE*: Extending BMA-AODE to Sets of Probabilities

By BMA-AODE* we extend BMA-AODE to *imprecise probabilities* (Walley, 1991), allowing multiple specifications of the prior mass function $P(S)$; we denote the *credal* set containing such prior mass functions as $\mathcal{P}(S)$. While a uniform prior represents prior *indifference* between the different SPODEs, the credal set represents a condition of prior *ignorance* about S, letting the prior probability of each SPODE vary within a large range. In principle we could let the prior probability of each SPODE vary exactly between zero and one (*vacuous* model). Yet, this would generate vacuous posterior inferences, thus preventing learning from data (Piatti et al., 2009). To obtain non-vacuous posterior inferences, we introduce non-zero lower bounds for the prior probability of

8

Algorithm 1 Robust exponentiation of log-likelihoods.

Require: Array `log_liks` of log-likelihoods, assumed of length `k`.

```
minVal=min(log_liks)

for i = 1:k do
    shifted_logliks(i)=logliks(i)-minVal;
    tmp_liks(i)=exp(shifted_logliks(i));
end for

total=sum(tmp_liks)

for i = 1:k do
    liks(i)=tmp_liks(i)/total;
end for

return liks {Array proportional to the exponentiated likelihoods}
```

---

the models. The resulting credal set is defined by the following constraints:

$$
\mathcal{P}(S) := \left\{ P(S) \,\middle|\, \begin{array}{ll} P(s_j) \geq \epsilon & \forall j = 1, \ldots, k \\ \sum_{j=1}^{k} P(s_j) = 1 \end{array} \right\}. \tag{4}
$$

The prior probability of each SPODE varies thus between ϵ and $1 - (k-1)\epsilon$. The set of mass functions in Eq.(4) is convex; its k extreme mass functions are those assigning mass ϵ to all the SPODEs apart from a single one, to which $1-(k-1)\epsilon$ is assigned. The constant ϵ appears in other places of this paper; in the implementation we set $\epsilon = 0.01$ for all occurrences of ϵ.

The credal set in (4) models the fact that, before observing the data, we are ignorant about the probability of each SPODE to be the true model. Considering that $\mathcal{P}(S)$ is a set a prior mass functions, BMA-AODE* can be regarded as a set of BMA-AODE classifiers, each corresponding to a different prior. The most probable class of an instance might happen to vary, when all the different priors of the credal set are considered; in this case the classification is *prior-dependent*. When dealing with prior-dependent instances, credal classifiers (Corani et al., 2012; Corani and Zaffalon, 2008b) become *indeterminate*, by returning a set of classes instead of a single class.

Before discussing how this set of classes is identified, let us introduce the concept

of *credal dominance* (or, for short, *dominance*): class c' *dominates* class c'' if c' is more probable than c'' under each prior of the credal set. If no class dominates c', then c' is non-dominated. Credal classifiers return in particular all the *non-dominated* classes, identified performing different by pairwise dominance tests among classes. This criterion is called *maximality* (Walley, 1991, Section 3.9.2) and is described by Algorithm 2. We point the reader to (Troffaes, 2007) for a discussion of alternative criteria for taking decisions under imprecise probabilities.

Algorithm 2 Identification of the non-dominated classes $\mathcal{ND}$ through maximality

---

$\mathcal{ND} := \mathcal{C}$

for $c' \in \mathcal{C}$ **do**
  for $c'' \in \mathcal{C}$ $(c'' \neq c')$ **do**

    check whether c' dominates c''

    if c' dominates c'' **then**
      remove c'' from $\mathcal{ND}$
    end if

  end for
end for

return $\mathcal{ND}$

Non-dominated classes are incomparable, namely there is no available information to rank them. Credal classifiers can be thus seen as dropping the dominated classes and expressing indecision about the non-dominated ones.

Within BMA-AODE*, c' dominates c'' if the solution of the following optimization problem is greater than one:

$$\text{minimize:} \quad \frac{\sum_{j=1}^{k} P(c'|\tilde{\mathbf{a}}, s_j) \cdot L_j \cdot P(s_j)}{\sum_{j=1}^{k} P(c''|\tilde{\mathbf{a}}, s_j) \cdot L_j \cdot P(s_j)}$$

$$\text{subject to:} \quad P(s_j) \geq \epsilon \quad \forall j = 1, \ldots, k$$

$$\sum_{j=1}^{k} P(s_j) = 1,$$

Note that the constrains of the problem correspond to the definition of credal set given in Eq.4. The above optimization task is a *fractional-linear program*; it can be mapped into a linear program by the Charnes-Cooper transformation (see Appendix Appendix A) and then solved *exactly*.

As already discussed for BMA-AODE, we include in the computation only the SPODEs whose conditional likelihood is at least $L_{\max}/10^4$. This can be regarded as a belief revision process, involving the credal set. The marginal credal set $\mathcal{P}'(Y)$ induces the following revision of the joint credal set $\mathcal{P}(X, Y)$:

$$\mathcal{P}'(X, Y) := \left\{ P'(X, Y) \middle| \begin{array}{l} P'(x, y) := P(x|y) \cdot P'(y) \\ P'(Y) \in \mathcal{P}'(Y) \end{array} \right\}.$$

It is worth emphasizing that the prior credal of BMA-AODE* includes the uniform prior adopted by BMA-AODE; therefore, the set of non-dominated classes identified by BMA-AODE* includes the most probable class returned by BMA-AODE; if in particular BMA-AODE* returns a single non-dominated class, this coincides to the class returned by BMA-AODE.

2.4. Compression-Based averaging

Compression-based averaging has been introduced by (Boullé, 2007) as a remedy against the tendency of BMA at getting excessively concentrated around the most probable model, which indeed deteriorates the performances (Boullé, 2007; Domingos, 2000). This approach replaces the posterior probabilities $P(s_j|\mathcal{D})$ of the models by smoother *compression* weights, which we denote as $P'(s_j|\mathcal{D})$ for model s_j. Note that also the adoption of the compression coefficients in place of the posterior probabilities can be seen as a belief revision.

To present the method, we need some further notation. In particular, we denote by LL_j the log of the conditional likelihood of model s_j. We moreover introduce the *null classifier* as a Bayesian network with no arcs, which models the class as independent from the features and whose probabilistic classifications correspond to the marginal probabilities of the classes. The null classifier will be used for the computation of the compression coefficients. We denote the null classifier as s_0; therefore we associated a further state s_0 to S, whose domain thus becomes $\{s_0, s_1, \ldots, s_k\}$. We denote as LL_0

the conditional log-likelihood of the null classifier. It has been shown (Boullé, 2007) that $LL_0 = -nH(C)$, where $H(C) := -\sum_{c \in \mathcal{C}} P(c) \log P(c)$ is the entropy[2] of the class.

Since we are dealing with a traditional single-prior classifier, we set a single prior mass function over the models, assigning uniform prior probability to the various SPODEs but prior probability ϵ to the null model; assigning a prior probability to the null model is necessary, since its posterior probability appears in the compression coefficients. Thus, we define the prior over variable S as follows:

$$P(s_j) = \begin{cases} \epsilon & j = 0, \\ \frac{1-\epsilon}{k} & j = 1, \ldots, k. \end{cases} \tag{5}$$

The compression coefficients are computed in two steps: computation of the *raw* compression coefficients and normalization. The *raw* compression coefficient associated to SPODE s_j is:

$$\pi_j := 1 - \frac{\log P(s_j|\mathcal{D})}{\log P(s_0|\mathcal{D})} = 1 - \frac{LL_j + \log P(s_j)}{LL_0 + \log P(s_0)} = 1 - \frac{LL_j + \log \frac{1-\epsilon}{k}}{-nH(C) + \log \epsilon}, \tag{6}$$

A negative π_j means that s_j is a worse predictor than the null model; a positive π_j means that s_j is a better predictor than the null model, which is the general case in practical situations. The upper limit of π_j is one: in this case s_j is a perfect predictor, with likelihood 1, and thus log-likelihood 0. Following (Boullé, 2007), we keep in the ensemble only the *feasible* models, namely those with $\pi_j > 0$; we instead discard the models with $\pi_j < 0$. Also this procedure corresponds to a belief revision induced by the removal from the ensemble of the models whose posterior probability falls below a certain threshold. Note also that, since $\pi_0 = 0$ by definition, the null model is not part of the resulting ensemble. The compression coefficients can be justified as follows (Boullé, 2007): $LL_j + \log P(s_j)$ "*represents the quantity of information required to encode the model plus the class values given the model. The code length of the null model can be interpreted as the quantity of information necessary to describe the classes, when no explanatory data is used to induce the model. Each model can poten-*

---

[2]For this equivalence to hold, it is necessary computing the entropy using the natural logarithm, instead of the $\log_2$ as usual.

tially exploit the explanatory data to better compress the class conditional information. The ratio of the code length of a model to that of the null model stands for a relative gain in compression efficiency."

With no loss of generality, assume the features to be ordered so that $A_1, A_2, \ldots, A_{\tilde{k}}$ yield a feasible model when used as super-parent; thus, SPODEs $s_1, s_2, \ldots, s_{\tilde{k}}$ are feasible, while SPODEs s_j with $j > \tilde{k}$ are removed from the ensemble. The *normalized* compression coefficients $P'(s_j|\mathcal{D})$ are obtained by normalizing the raw compression coefficients of the feasible SPODEs:

$$P'(s_j|\mathcal{D}) = \begin{cases} \dfrac{\pi_j}{\sum_{l=1}^{\tilde{k}} \pi_l} & \text{if } j = 1, \ldots, \tilde{k}, \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

The posterior probabilities are estimated as:

$$P(c|\tilde{\mathbf{a}}) = \sum_{j=1}^{k} P(c|\tilde{\mathbf{a}}, s_j) \cdot P'(s_j|\mathcal{D}). \tag{8}$$

We call this classifier COMP-AODE, where COMP stands for compression-based. COMP-AODE performs a weighted linear combination of probabilities estimated by different models; in risk analysis, a weighted linear combination of probabilities estimated by different experts is referred to as *linear opinion pool* (Clemen and Winkler, 1999).

2.5. COMP-AODE*: Extending COMP-AODE to Sets of Probabilities

We extend COMP-AODE to imprecise probabilities by allowing for multiple specifications of the prior $P(S)$ over the models, collected into a credal set $\mathcal{P}_c(S)$, where the subscript denotes compression. Differently from the credal set $\mathcal{P}(S)$ used by the BMA-AODE*, here we also consider the null model. We assign to the null model a fixed prior probability ϵ, while the prior probability of the SPODEs are free to vary under constraints analogous to those of BMA-AODE*; in this way we model a condition of prior *ignorance*. The credal set $\mathcal{P}_c(S)$ adopted by COMP-AODE* is therefore:

$$\mathcal{P}_c(S) := \left\{ P(S) \middle| \begin{array}{l} P(s_0) = \epsilon, \\ P(s_j) \geq \epsilon \quad \forall j = 1, \ldots, k, \\ \sum_{j=0}^{k} P(s_j) = 1 \end{array} \right\}. \tag{9}$$

13

The bounds of the raw compression coefficient defined in Eq.(6) are obtained by letting vary $P(S)$ in $\mathcal{P}_c(S)$:

$$\pi_j \in \left[1 - \frac{LL_j + \log \epsilon}{-nH(C) + \log \epsilon}, \ 1 - \frac{LL_j + \log (1 - k\epsilon)}{-nH(C) + \log \epsilon} \right]. \tag{10}$$

Since the prior used by COMP-AODE (Eq. 5) belongs to the credal set of COMP-AODE*, the point estimate of the compression coefficient adopted by COMP-AODE (Eq.6) lies in the above interval. Note also that the upper bound of the above interval (*upper coefficient of compression*) is obtained in correspondence of the *extreme* mass function which assigns prior probability $1 - k\epsilon$ to model s_j and prior probability ϵ to all the remaining models. The various π_j cannot vary independently from each other; they are instead linked by the normalization constraint in Eq.(9).

COMP-AODE* regards SPODE s_j as non-feasible if the *upper* coefficient of compression is non-positive: this approach thus preserves all the models which are feasible, in the sense of Section 2.4, for at least a prior in the set $\mathcal{P}_c(S)$. COMP-AODE* is thus more conservative than COMP-AODE, namely it discards a lower number of models. However, generally neither COMP-AODE* nor COMP-AODE remove any SPODE from the ensemble. Since the prior adopted by COMP-AODE is contained in the credal set of COMP-AODE*, the most probable class identified by COMP-AODE is part of the non-dominated classes identified by COMP-AODE*. [3]

Like BMA-AODE*, COMP-AODE* identifies for each instance the non-dominated classes through maximality (Algorithm 2). In the following, we explain how to compute the test of dominance among two classes.

Testing dominance

Without loss of generality, we assume the features to have be re-ordered, so that the first $\tilde{k}$ features yield a model with positive *upper* coefficient of compression when used as super-parent; in other words, SPODEs $\{s_1, \ldots, s_{\tilde{k}}\}$ are the feasible ones. In this case the dominance test corresponds to evaluate whether or not the solution of the following optimization problem is greater than one.

---

[3]Exception to this statement are in principle possible if the set of feasible SPODEs differs between COMP-AODE* and COMP-AODE. However, this did not happen in our extensive experiments.

$$\text{minimize:} \quad \frac{P(c'|\mathbf{a})}{P(c''|\mathbf{a})} \propto \frac{\sum_{j=1}^{\tilde{k}} P(c'|\tilde{\mathbf{a}}, s_j) \cdot \pi_j}{\sum_{j=1}^{\tilde{k}} P(c''|\tilde{\mathbf{a}}, s_j) \cdot \pi_j} \tag{11}$$

$$\text{w.r.t.:} \quad P(s_0), P(s_1), \ldots, P(s_k) \tag{12}$$

$$\text{subject to:} \quad P(s_0) = \epsilon \tag{13}$$

$$P(s_j) \geq \epsilon \quad \forall j = 1, \ldots, k$$

$$\sum_{j=1}^{k} P(s_j) = 1,$$

where the normalization term $\sum_{j=1}^{\tilde{k}} \pi_j$ has been already simplified, being positive by definition. Recalling that $P(s_0) = \epsilon$ and introducing Eq.(6) which shows how π_j depends on the optimization variable $P(s_j)$, we rewrite the objective function as:

$$\frac{\sum_{j=1}^{\tilde{k}} P(c'|\mathbf{a}, s_j) \cdot \left(1 - \frac{\log P(s_j) + LL_j}{\log \epsilon + LL_0}\right)}{\sum_{j=1}^{\tilde{k}} P(c''|\mathbf{a}, s_j) \cdot \left(1 - \frac{\log P(s_j) + LL_j}{\log \epsilon + LL_0}\right)},$$

and hence

$$\frac{\sum_{j=1}^{\tilde{k}} P(c'|\mathbf{a}, s_j) \cdot (\log \epsilon + LL_0 - LL_j) - \sum_{j=1}^{\tilde{k}} P(c'|\mathbf{a}, s_j) \cdot \log P(s_j)}{\sum_{j=1}^{\tilde{k}} P(c''|\mathbf{a}, s_j) \cdot (\log \epsilon + LL_0 - LL_j) - \sum_{j=1}^{\tilde{k}} P(c''|\mathbf{a}, s_j) \cdot \log P(s_j)}.$$

We then introduce the constants $a := \sum_{j=1}^{\tilde{k}} P(c'|\mathbf{a}, s_j) (\log \epsilon + LL_0 - LL_j)$, $b := \sum_{j=1}^{j=\tilde{k}} P(c''|\mathbf{a}, s_j) (\log \epsilon + LL_0 - LL_j)$, $\alpha_j := P(c'|\mathbf{a}, s_j)$, $\beta_j := P(c''|\mathbf{a}, s_j)$. After changing the sign of both numerator and denominator of the objective function, we rewrite the optimization problem, with respect to the variables $x_1, x_2, \ldots, x_{\tilde{k}}$, where $x_j := \log P(s_j)$, as follows:

$$\text{minimize:} \quad \frac{\sum_{j=1}^{\tilde{k}} \alpha_j x_j - a}{\sum_{j=1}^{\tilde{k}} \beta_j x_j - b}$$

$$\text{w.r.t.:} \quad x_1, \ldots, x_{\tilde{k}}$$

$$\text{subject to:} \quad x_j \geq \log \epsilon \quad \forall j = 1, \ldots, \tilde{k},$$

15

$$\sum_{j=1}^{\tilde{k}} \exp x_j = 1 - \epsilon - (k - \tilde{k})\epsilon.$$

where the constrains are derived from the definition of credal set (9). The last constraint is justified as follows: $(k - \tilde{k})$ models have been removed from the ensemble as unfeasible and therefore they do not appear in the optimization problem. Without changing the credal set, we set their priors to ϵ; since these models do not impact on the objective function, the best solution is attained by allocating to them the minimum possible prior probability. We then substitute $y_j := \exp x_j$ to avoid numerical problems in the optimization, thus getting the following non-linear optimization problem with linear constraints:

$$\text{minimize:} \qquad \frac{\sum_{j=1}^{\tilde{k}} \alpha_j \cdot \log y_j - a}{\sum_{j=1}^{\tilde{k}} \beta_j \cdot \log y_j - b} \tag{14}$$

$$\text{w.r.t.:} \qquad y_1, \ldots, y_{\tilde{k}} \tag{15}$$

$$\text{subject to:} \qquad y_j \geq \epsilon, \tag{16}$$

$$\sum_{j=1}^{\tilde{k}} y_j = 1 - (k - \tilde{k} - 1)\epsilon.$$

2.6. Computational Complexity of the Classifiers

We now analyze the computational complexity of the proposed classifiers and compare it with that of the standard AODE. We distinguish between *learning* and *classification* complexity, the latter referring to the classification of a single instance. Both the *space* and the *time* required for computations are evaluated. The orders of magnitude of these descriptors are reported as a function of the dataset size n, the number of attributes/SPODEs k, the number of classes $l := |\mathcal{C}|$, and average number of states for the attributes $v := k^{-1} \sum_{i=1}^{k} |\mathcal{A}_i|$. A summary of this analysis is in Table 1 and the discussion below.

Let us first evaluate the AODE. For a single SPODE s_j, the tables $P(C)$, $P(A_j|C)$ and $P(A_i|C, A_j)$, with $i = 1, \ldots, k$ and $i \neq j$ should be stored, this implying space complexity $\mathcal{O}(lkv^2)$ for learning each SPODE and $\mathcal{O}(lk^2v^2)$ for the AODE ensemble.

Algorithm	Space	Time	
	learning/classification	learning	classification
AODE	$\mathcal{O}(lk^2v^2)$	$\mathcal{O}(nk^2)$	$\mathcal{O}(lk^2)$
BMA-AODE/COMP-AODE	$\mathcal{O}(lk^2v^2)$	$\mathcal{O}(n(l+k)k)$	$\mathcal{O}(lk^2)$
BMA-AODE*/COMP-AODE*	$\mathcal{O}(lk^2v^2)$	$\mathcal{O}(n(l+k)k)$	$\mathcal{O}(l^2k^3)$

Table 1: Complexity of classifiers.

These tables should be available during learning and classification for both classifiers; thus, space requirements of these two stages are the same.

Time complexity to scan the dataset and learn the probabilities is $\mathcal{O}(nk)$ for each SPODE, and hence $\mathcal{O}(nk^2)$ for the AODE. The time required to compute the posterior probabilities is $\mathcal{O}(lk)$ for each SPODE, and hence $\mathcal{O}(lk^2)$ for AODE.

Learning BMA-AODE or COMP-AODE takes the same space as AODE, but higher computational time, due to the evaluation of the conditional likelihood as in Eq.(2). The additional computational time is $\mathcal{O}(nlk)$, thus requiring $\mathcal{O}(n(l+k)k)$ time overall. For classification, time and space complexity during learning and classification are just the same.

The credal classifiers BMA-AODE* and COMP-AODE* require the same space complexity and the same time complexity in learning of their non-credal counterparts. However, credal classifiers have higher time complexity in classification. The pair-wise dominance tests in Algorithm 2 requires the solution of a number of optimization problems for each test instance which is quadratic in the number of classes. We can roughly describe as cubic in the number of variables the time complexity of solving the linear programming problem for BMA-AODE* and the optimization of the non-linear function, with linear constraints, for COMP-AODE*. Summing up credal classifiers increase of one unit, compared to their single-prior counterparts, the exponents of the number of classes and attributes in the time complexity of the classification stage.

3. Experiments

We run experiments on 40 data sets, whose characteristics are given in the Appendix (Table B.2). On each data set we perform 10 runs of 5-fold cross-validation. In order to have complete data, we replace missing values with the median and the mode

for respectively numerical and categorical features. We discretize numerical features by the entropy-based method of (Fayyad and Irani, 1993). For pairwise comparison of of classifiers over the collection of data sets we use the non-parametric Wilcoxon signed-rank test.[4] The Wilcoxon signed-rank test is indeed recommended for comparing two classifiers on multiple data sets (Demsar, 2006): being non-parametric it avoids strong assumptions and robustly deals with outliers.

3.1. Determinate classifiers

We call *determinate* the classifiers which always return a single class, namely AODE, BMA-AODE and COMP-AODE. For determinate classifiers we use two indicators: the accuracy, namely the percentage of correct classifications, and the Brier loss

$$\frac{1}{n_{te}} \sum_{i}^{n_{te}} \left(1 - P(c^{(i)}|\mathbf{a}^{(i)})\right)^2,$$

where n_{te} denotes the number of instances in the test set, while $P(c^{(i)}|\mathbf{a}^{(i)})$ is the probability estimated by the classifier for the true class of the i-th instance. The Brier loss assesses the quality of the estimated probabilities in a more sensitive way than accuracy.

A first finding is that AODE outperforms BMA-AODE, having both higher accuracy (p-value $< .01$) and lower Brier loss. We present in Figure 3(a) the scatter plot of accuracies and in Figure 4(a) the *relative* Brier losses, namely the Brier loss of BMA-AODE divided, data set by data set, by the Brier loss of AODE. On average, BMA-AODE has 3% higher Brier loss than AODE. The fact that AODE outperforms BMA-AODE could be expected; the same finding was already given in (Yang et al., 2007) and in (Cerquides et al., 2005), with the main difference that the BMA-AODE of these works was based on the marginal likelihood rather than on the conditional likelihood. Our results show that BMA-AODE is outperformed by AODE, even when using the conditional likelihood. BMA-AODE is outperformed by AODE both because its excessive concentration around the most probable model (Boullé, 2007; Cerquides et al.,

---

[4]For each indicator of performance, we generate two *paired* vectors: the same position in both vectors refers to the same data set. The two vectors are then used as input for the test.

2005; Domingos, 2000; Minka, 2002) which tends to cancel the advantage of averaging over models, and because of the effectiveness of simply averaging over SPODEs, as done by AODE, in terms of reduction of the variance error.

As outlined by Figure 3(b), the difference between accuracies is instead not significant when comparing COMP-AODE and AODE. However COMP-AODE outperforms AODE on the Brier loss (p-value $< .01$); in Figure 4(b) we show the *relative* Brier losses, namely the Brier loss of COMP-AODE divided, data set by data set, by the Brier loss of AODE. Averaging over data sets, COMP-AODE reduces the Brier loss of about 3% compared to AODE. We see this result as noteworthy, since AODE is a high performance classifier. These positive results with the compression-based approach broaden the scope of the experiments of (Boullé, 2007), in which the compression approach was applied to an ensemble of naive Bayes classifiers.
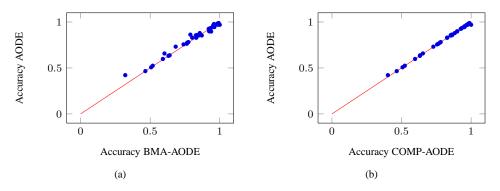

(a)   (b)

Figure 3: Scatter plots of accuracies; the solid line shows the bisector.

3.2. Credal classifiers

A credal classifier can be seen as separating the instances into two groups: the *safe* ones, for which it returns a single class is returned, and the *prior-dependent* ones, for which it returns two or more classes. Note that prior-dependence is not an intrinsic property to the instance: an instance can be judged as prior-dependent by a certain credal classifier and as safe by a different credal classifier. To characterize the performance of a credal classifier, the following four indicators are considered (Corani and Zaffalon, 2008b):
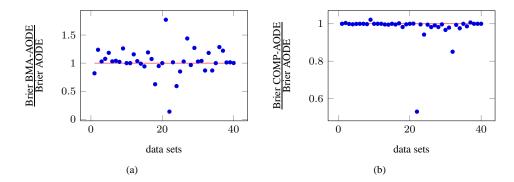
Figure 4: Relative Brier losses; points lying *below* the horizontal line represent performance better than AODE, and vice versa. Note the different y-scales of the two graphs.

- *determinacy*: % of instances recognized as safe, namely classified with a single class;

- *single-accuracy*: the accuracy achieved over the instances recognized as safe;

- *set-accuracy*: the accuracy achieved, by returning a set of classes, over the prior-dependent instances;

- *indeterminate output size*: the average number of classes returned on the prior-dependent instances.

Averaging over data sets, BMA-AODE* has 94% determinacy; it is completely determinate on 7 data sets. However, this determinacy fluctuates among data sets, showing for instance a significant correlation with the sample size n ($\rho = 0.3$). The choice of the prior is less important on large data sets: bigger data sets tend to contain a lower percentage of prior-dependent instances, thus increasing determinacy. BMA-AODE* performs well when indeterminate: averaging over all data sets, it achieves 90% set-accuracy by returning 2.3 classes (the average number of classes in the collection of data sets is 3.6). It is worth analyze the performance of BMA-AODE on the prior-dependent instances. In Figure 5(a) we compare, data set by data set, the accuracy achieved by BMA-AODE on the instances judged respectively as safe and as prior-dependent by BMA-AODE*; the plot shows a sharp drop of accuracy on the prior-dependent instances, which is statistically significant (p-value $< .01$). As a rough

20

indication, averaging over data sets, the accuracy of BMA-AODE is 83% on the safe instances but only 52% on the instances recognizes as prior-dependent by BMA-AODE*. Thus, on the prior-dependent instances, BMA-AODE provides fragile classifications; on the same instances, BMA-AODE* returns a small-sized but highly accurate set of classes.
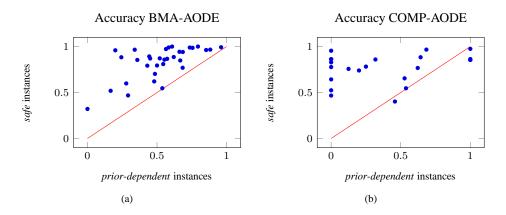

Figure 5: Accuracy of the determinate classifiers on the instances recognized as safe and as prior-dependent by their credal counterparts. The accuracies of BMA-AODE [COMP-AODE] is thus separately measured on the instances judged safe and prior-dependent by BMA-AODE* [COMP-AODE*]. The solid line shows the bisector.

Let us now analyze the performance of COMP-AODE*; it has higher determinacy than BMA-AODE*; averaging over data sets, its determinacy is 99%, with only minor fluctuations across data sets; the classifier is moreover completely determinate on 18 data sets. The determinacy of COMP-AODE* is very high and stable across data sets. Therefore, under the compression-based approach only a small fraction of the instances is prior-dependent; this robustness to the choice of the prior is likely to contribute to the good performance of compression-based ensemble of classifiers and constitutes a desirable but previously unknown property of the compression-based approach. Numerical inspection shows that the logarithmic smoothing of the models' posterior probabilities makes indeed the compression weights only little sensitive to the choice of the prior. COMP-AODE* performs well when indeterminate: averaging over all data sets, it achieves 95% set-accuracy by returning 2 classes (note that the indeterminate output size cannot be less than two).

Again, it is worth checking the behavior of the corresponding determinate classifier, namely COMP-AODE, on the instances that are prior-dependent for the COMP-AODE*. In Figure 5(b) we compare, data set by data set, the accuracy achieved by COMP-AODE on the instances judged respectively safe and prior-dependent by COMP-AODE*; there is a large drop of accuracy on the prior-dependent instances, and the drop is significant (p-value $< .01$). Averaging over data sets, the accuracy of COMP-AODE drops from 82% on the safe instances to only 47% on the instances judged as prior-dependent by COMP-AODE*. Even COMP-AODE, despite its robustness to the specification of the prior, undergoes a severe loss of accuracy on the instances recognized as prior-dependent by COMP-AODE*. On the very same instances, COMP-AODE* returns a small sized but highly reliable set of classes, thus enhancing the overall classification reliability.

3.3. Utility-based Measures

We have seen so far that the credal classifiers extend in a sensible way their determinate counterparts, being able to recognize prior-dependent instances and to robustly deal with them. Yet, it is not obvious how to compare credal and determinate classifiers by means of a synthetic indicator. In fact, to fairly compare determinate and indeterminate predictions is very challenging; to the best of our knowledge, a satisfactory solution exists only for 0-1 loss, while comparing determinate and indeterminate predictions in a cost-sensitive setting, in which different kind of errors imply different costs, is still an open problem. In the following we thus reason under 0-1 loss. The *discounted accuracy* rewards a prediction made of m classes with $1/m$ if it contains the true class, and with 0 otherwise. Discounted accuracy is then compared to the accuracy achieved by a determinate classifier. A theoretical justification for discounted-accuracy has been given by Zaffalon et al. (2011) showing that, within a betting framework based on fairly general assumptions, discounted-accuracy is the only score which satisfies some fundamental properties for assessing both determinate and indeterminate classifications. Yet Zaffalon et al. (2011) also shows some severe limits of discounted-accuracy, which we illustrate by means of an example: we consider two different medical doctors, doctors *random* and doctor *vacuous*, who should decide whether a patient is *healthy* or

diseased. Doctor *random* issues random diagnosis, using a uniform distribution over the two categories. Doctor *vacuous* instead always return both categories, admitting to be ignorant. Let us assume that the hospital profits a quantity of money proportional to the discounted-accuracy achieved by its doctors at each visit. Both doctors have the same *expected* discounted-accuracy for each visit, namely $1/2$. For the hospital, both doctors provide the same *expected* profit on each visit, but with a substantial difference: the profit of doctor vacuous is *deterministic*, while the profit of doctor random is affected by considerable variance. Any risk-averse hospital manager should thus prefer doctor vacuous over doctor random, since it yields the same expected profit with less variance. In fact, under risk-aversion, the expected utility increases with expectation of the rewards and decreases with their variance (Levy and Markowitz, 1979). To capture this point it is necessary introducing a utility function, to be then applied on the discounted-accuracy score assigned on each instance. In Zaffalon et al. (2011) the utility function is designed as follows: the utility of a correct and determinate classification (discounted-accuracy 1) is 1; the utility of a wrong classification (discounted-accuracy 0) is 0; the utility of an accurate but indeterminate classification consisting of two classes (discounted-accuracy 0.5) is assumed to lie between 0.65 and 0.8. Two quadratic utility functions are then derived corresponding to these boundary values, and passing respectively through $\{u(0) = 0, u(0.5) = 0.65, u(1) = 1\}$ and $\{u(0) = 0, u(0.5) = 0.8, u(1) = 1\}$, denoted as u_{65} and u_{80} respectively[5]. Since $u(1) = 1$, utility and accuracy coincide for determinate classifiers; therefore, utility of credal classifiers and accuracy of determinate classifiers can be directly compared. In del Coz and Bahamonde (2009) classifiers which return indeterminate classifications are scored through the F_1-metric, originally designed for Information Retrieval tasks. The F_1 metric, when applied to indeterminate classifications, returns a score which is always comprised between u_{65} and u_{80}, further confirming the reasonableness of both utility functions. More details on the links between F_1, u_{65} and u_{80} are given in Zaffalon et al. (2012). We remark that in real applications the utility function should

---

[5]The mathematical expression of these utility functions are as follows: $u_{65}(x) = -1.2x^2 + 2.2x$, $u_{80}(x) = -0.6x^2 + 1.6x$, where x is the value of discounted accuracy.

be elicited by discussion with the decision maker; in this paper we use u_{65} and u_{80} to model two reasonable but different degrees of risk-aversion.
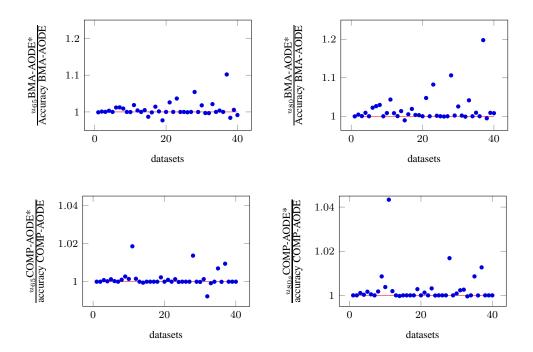

Figure 6: Relative utilities of credal classifiers compared to their precise counterparts.

We now analyze the utilities generated by the various classifiers, comparing each credal classifier with its determinate counterpart. BMA-AODE* has significantly higher utility (p-value $< .01$) than BMA-AODE under both u_{65} and u_{80}. This confirms that extending the model to imprecise probability is a sensible approach. In the first row of Figure 6 we show the *relative* utility, namely the utility of BMA-AODE* divided, data set by data set, by the utility (i.e., accuracy) of BMA-AODE; the two plots refer respectively to u_{65} and u_{80}. Averaging over data sets, the improvement of utility is about 1% and 2% under u_{65} and u_{80}; although the improvement might look small, we recall that it is obtained by modifying the classifications of the prior-dependent instances only, 6% of the total on average. If we focus on the prior-dependent instances only, the increase of utility generally varies between +10% and +40% depending on the data set and on the utility function. Clearly, the improvement is even larger under u_{80} which

24

assigns higher utility than u_{65} to the indeterminate but accurate classifications.

The analysis is similar when comparing COMP-AODE* with COMP-AODE. In the second row of Figure 6 we show the *relative* utility, namely the utility of COMP-AODE* divided, data set by data set, by the utility (i.e., accuracy) of COMP-AODE. The increase of utility is in this case generally under 1%, as a consequence of the higher determinacy of COMP-AODE (99% on average), which allows less room for improving utility through indeterminate classifications. In fact, the robustness of COMP-AODE to the choice of the prior reduces the portion of instances where it is necessary making the classification indeterminate. Focusing however on the (rare) indeterminate instances, the increase of utility deriving to the extension to imprecise probability lies between 39% and 60%, depending on the data set and on the utility function. Eventually, COMP-AODE* has significantly (p-value $< .01$) higher utility than COMP-AODE under *both* u_{65} and u_{80}; also in this case the extension to the credal paradigm is beneficial.

The utilities of COMP-AODE* and BMA-AODE* are also compared; under u_{65} COMP-AODE* yields significantly (p-value $< .05$) higher utility than BMA-AODE*, while under u_{80} the difference among the two classifiers is not significant, although the utility generated by COMP-AODE* is generally slightly higher. The point is that BMA-AODE* is more often indeterminate than COMP-AODE*; under u_{80} the indeterminate but accurate classifications are rewarded more than under u_{65}, thus allowing BMA-AODE* to almost close the gap with COMP-AODE*. We conclude however that COMP-AODE* should be generally preferred over BMA-AODE*.

Eventually we point out that COMP-AODE* generates significantly (p-value $< .01$) higher utility than AODE, under *both* u_{65} and u_{80}. The extension to imprecise probability has thus concretely improved the overall performance of the compression-based ensemble: recall that the determinate COMP-AODE yields better probability estimates but not better accuracy than AODE.

3.4. Comparison with previous credal classifiers

In this section we compare COMP-AODE* with previous credal classifiers. A well-known credal classifier is the *naive credal classifier* (NCC) (Corani and Zaffalon,
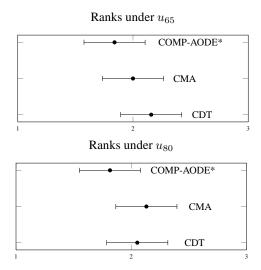
Figure 7: Comparison between credal classifiers by means of the Friedman test: the boldfaced points show the average ranks; a lower rank implies better performance. The bars display the critical distance, computed with 95% confidence: the performance of two classifiers are significantly different if their bars do not overlap.

2008b), which is an extension of naive Bayes to imprecise probability. We have ran NCC on the same collection of data sets following the experimental setup of Section 3; under *both* u_{65} and u_{80}, the utility produced by COMP-AODE* is significantly higher ($p <$0.01) than that produced by NCC. Thus, COMP-AODE* outperforms NCC.

However, over time algorithms more sophisticated than NCC have been developed, such as:

- *credal model averaging* (CMA) (Corani and Zaffalon, 2008a), namely a generalization of BMA (in the same spirit of BMA-AODE) for naive Bayes classifier;

- *credal decision tree* (CDT) (Abellán and Moral, 2005), namely an extension of classification trees to imprecise probability.

We then compare CDT, CMA and COMP-AODE* via the Friedman test; this is the approach recommended by (Demsar, 2006) for comparing multiple classifiers on multiple data sets. First, the procedure ranks on each data set the classifiers according to the utility they generate; then, it tests the null hypothesis of all classifiers having the same average rank across the data sets. If the null hypothesis is rejected, a post-hoc test is adopted to identify the significant differences among classifiers. Adopting a 95% con-

26

fidence, no significant difference is detected among classifiers; the result is the same under both utilities. However, under both utilities COMP-AODE* has the best average rank, as shown in Figure 3.4. Lowering the confidence to 90%, two significant differences are found: a) COMP-AODE* produces significantly higher utility than CMA under u_{65} and b) COMP-AODE* produces significantly higher utility than CDT under u_{80}. These results, though not completely conclusive, suggest that COMP-AODE* compares favorably to previous credal classifiers.

3.5. *Some comments on credal classification vs reject option*

Determinate classifiers can be equipped with a *reject option* (Herbei and Wegkamp, 2006), thus refusing to classify an instance if the posterior probability of the most probable class is less than a threshold. For the sake of simplicity we consider a case with two classes only; to formally introduce the reject option, it is necessary setting a cost d ($0 < d < 1/2$), which is incurred when rejecting an instance. A cost 0, 1, d is therefore incurred when respectively correctly classifying, wrongly classifying and rejecting an instance. Under 0-1 loss, the *expected* cost for classifying an instance corresponds to the probability of misclassification; it is thus $1 - p^*$, where p^* denotes the posterior probability of the most probable class. The optimal behavior is thus to reject the classification whenever the expected classification cost is higher than the rejection cost, namely when $(1 - p^*) > d$; this is equivalent to rejecting the instance whenever $p^* < 1 - d$, where $(1 - d)$ constitutes the *rejection threshold*.

The behavior induced by the reject option is quite different from that of a credal classifier, as we show in the following example. On an a very large data set the posterior probability of the classes is little sensitive on the choice of the prior, because of the wide amount of data available for learning; in this condition, instance are rarely prior-dependent and therefore a credal classifier will mostly return a single class. On the other hand, the determinate classifier with reject option (RO in the following) rejects all the instances for which $p^* < 1 - d$; if d is small, there can be even a high number of rejected instances. The difference between these behaviors is due to the credal classifier being unaware of the cost d associated with rejecting an instance, which is instead driving the behavior of RO. To rigorously compare RO against a credal classifier, it is

thus necessary making the credal classifier aware of the cost d. Recalling that the credal classifier already returns both classes on the instances which are prior dependent, this will change the behavior of the credal classifier only on the instances which are *not* prior-dependent. In particular, the credal classifier should reject all the instances for which $\underline{p}^* < 1 - d$, where $\underline{p}^*$ is the *lower* probability of the most probable class; the instances rejected by means of this criterion will be thus a superset of those rejected by RO. Therefore, the credal classifier will reject the instances which are prior-dependent *and* those for which $\underline{p}^* < 1 - d$. Eventually, the cost generated by the credal classifier should be compared with those generated by the RO. In the case with more than 2 classes the analysis might become slightly more complicated than what discussed here; however, we leave the analysis of credal classifiers with reject option as a topic for future research. Note also that this kind of experiment will require the computation of upper and lower posterior probability of the classes, which is not always trivial with credal classifiers.

4. Conclusions

Applying Bayesian Model Averaging over SPODEs actually worsens the classification performance compared to the standard AODE. Instead the COMP-AODE classifier proposed here, which applies the compression-based approach over SPODEs, obtains overall slightly better classification performance than AODE; our results thus broadens the scope of (Boullé, 2007), in which the compression-based approach was applied over an ensemble of naive Bayes classifiers. The two credal classifiers BMA-AODE* and COMP-AODE* extend respectively BMA-AODE and COMP-AODE to imprecise probability, replacing the uniform prior over the SPODEs by a *credal set*; both credal classifiers automatically identify the prior-dependent instances, and cope reliably with them by returning a small-sized but highly accurate set of classes. On the prior-dependent instances both BMA-AODE and COMP-AODE undergo a severe drop of accuracy. Both BMA-AODE* and COMP-AODE* provide overall higher performance than their determinate counterparts as measured by the utility-based measures, which to our knowledge constitute the state of the art for comparing determinate

28

and credal classifiers. According to the same metrics, COMP-AODE* shows better performance than previous credal classifiers.

Acknowledgements

References

Abellán, J., Moral, S., 2005. Upper entropy of credal sets. applications to credal classification. International Journal of Approximate Reasoning 39 (2–3), 235–255.

Bajalinov, E., 2003. Linear-fractional programming: theory, methods, applications and software. Springer Netherlands.

Boullé, M., 2007. Compression-based averaging of selective naive bayes classifiers. Journal of Machine Learning Research 8, 1659–1685.

Cerquides, J., De Màntaras, R., et al., 2005. Robust Bayesian linear classifier ensembles. Lecture notes in computer science 3720, 72.

Clemen, R., Winkler, R., 1999. Combining probability distributions from experts in risk analysis. Risk Analysis 19 (2), 187–203.

Corani, G., Antonucci, A., Zaffalon, M., 2012. Bayesian networks with imprecise probabilities: Theory and application to classification. In: Holmes, D. E., Jain, L. C., Kacprzyk, J., Jain, L. C. (Eds.), Data Mining: Foundations and Intelligent Paradigms. Vol. 23. Springer Berlin Heidelberg, pp. 49–93.

Corani, G., Zaffalon, M., 2008a. Credal model averaging: an extension of Bayesian model averaging to imprecise probabilities. In: Daelemans, W., Goethals, B., Morik, K. (Eds.), Proc. 12th European Conference on Machine Learning (ECML-PKDD 2008). Springer, pp. 257–271.

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

Cowell, R., 2001. On searching for optimal classifiers among bayesian networks. In: Proceedings of the Eighth International Conference on Artificial Intelligence and Statistics. pp. 175–180.

Dash, D., Cooper, G., 2004. Model Averaging for Prediction with Discrete Bayesian Networks. Journal of Machine Learning Research 5, 1177–1203.

del Coz, J., Bahamonde, A., 2009. Learning nondeterministic classifiers. Journal of Machine Learning Research 10, 2273–2293.

Demsar, J., 2006. Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 7, 1–30.

Domingos, P., 2000. Bayesian averaging of classifiers and the overfitting problem. In: Proc. of the 17th International Conference on Machine Learning. pp. 223–230.

Domingos, P., Pazzani, M., 1997. On the optimality of the simple Bayesian classifier under zero-one loss. Machine Learning 29 (2/3), 103–130.

Dubois, D., Prade, H., 1997. A synthetic view of belief revision with uncertain inputs in the framework of possibility theory. International Journal of Approximate Reasoning 17, 295–324.

Fayyad, U. M., Irani, K. B., 1993. Multi-interval discretization of continuous-valued attributes for classification learning. In: Proceedings of the 13th international joint conference on artificial intelligence. Morgan Kaufmann, San Francisco, CA, pp. 1022–1027.

Friedman, J., 1997. On bias, variance, 0/1 - loss, and the curse-of-dimensionality. Data Mining and Knowledge Discovery 1, 55–77.

Friedman, N., Geiger, D., Goldszmidt, M., 1997. Bayesian networks classifiers. Machine Learning 29 (2/3), 131–163.

Herbei, R., Wegkamp, M., 2006. Classification with reject option. Canadian Journal of Statistics 34 (4), 709–721.

Hoeting, J., Madigan, D., Raftery, A., Volinsky, C., 1999. Bayesian Model Averaging: a Tutorial. Statistical Science 14 (4), 382–417.

Kontkanen, P., Myllymaki, P., Silander, T., Tirri, H., 1999. On supervised selection of bayesian networks. In: Proc. of the Fifteenth Conference on Uncertainty in Artificial Intelligence. pp. 334–342.

Levi, I., 1980. The Enterprise of Knowledge. MIT Press, London.

Levy, H., Markowitz, H., 1979. Approximating expected utility by a function of mean and variance. The American Economic Review 69 (3), 308–317.

Minka, T., 2002. Bayesian model averaging is not model combination. Tech. rep., MIT Media Lab note.
URL http://research.microsoft.com/~{}minka/papers/minka-bma-isnt-mc.pdf

Piatti, A., Zaffalon, M., Trojani, F., Hutter, M., 2009. Limits of learning about a categorical latent variable under prior near-ignorance. International Journal of Approximate Reasoning 50, 597–611.

Troffaes, M., 2007. Decision making under uncertainty using imprecise probabilities. International Journal of Approximate Reasoning 45 (1), 17–29.

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

Webb, G., Boughton, J., Wang, Z., 2005. Not so naive bayes: Aggregating one-dependence estimators. Machine Learning 58 (1), 5–24.

Yang, Y., Webb, G., Cerquides, J., Korb, K., Boughton, J., Ting, K., 2007. To select or to weigh: A comparative study of linear combination schemes for superparent-one-dependence estimators. Knowledge and Data Engineering, IEEE Transactions on 19 (12), 1652–1665.

Zaffalon, M., Corani, G., Mauá, D., 2011. Utility-based accuracy measures to empirically evaluate credal classifiers. In: Coolen, F., de Cooman, G., Fetz, T., Oberguggenberger, M. (Eds.), ISIPTA'11: Proceedings of the Seventh International Symposium on Imprecise Probability: Theories and Applications. SIPTA, Innsbruck, pp. 401–410.

Zaffalon, M., Corani, G., Maua, D., 2012. Evaluating credal classifiers by utility-discounted predictive accuracy. Tech. Rep. IDSIA-03-12, IDSIA (Istituto Dalle Molle Intelligenza Artificiale.

Appendix A. Mapping linear-fractional programs to linear programs by the Charnes-Cooper transformation

In this appendix, we adapt the classical Charnes-Cooper transformation to the particular linear-fractional program to be solved to test dominance for the BMA-AODE* as described in Section 2.3. Let us write the optimization variables as $x_j := P(s_j)$ (with $j = 1, \ldots, k$) and the coefficients as:

$$
\begin{bmatrix} \gamma_i \\ \delta_i \end{bmatrix} := \begin{bmatrix} P(c'|\tilde{\mathbf{a}}, m_j) \\ P(c''|\tilde{\mathbf{a}}, m_j) \end{bmatrix} \cdot L_j. \tag{A.1}
$$

The objective function rewrites therefore as:

$$
\frac{\sum_{j=1}^{k} \gamma_j x_j}{\sum_{j=1}^{k} \delta_j x_j}. \tag{A.2}
$$

with $j = 1, \ldots, k$. Let us indeed change the variables as follows:

$$
y_j := \frac{x_j}{\sum_j \delta_j x_j}, \tag{A.3}
$$

and introduce the auxiliary variable

$$
t := \frac{1}{\sum_j \delta_j x_j}. \tag{A.4}
$$

After this, non-linear, transformation, the objective function takes a linear form:

$$
\sum_j \gamma_j y_j, \tag{A.5}
$$

while each linear constraint $x_j \geq \epsilon$, rewrites as $y_j \geq \epsilon t$, thus being still linear. Similarly, the normalization rewrites as:

$$
\sum_j y_j = t.
$$

We have therefore mapped the original problem into a standard linear program and the solutions of the two problems are known to coincide (Bajalinov, 2003, Chap. 3). Note that the transformation only increases by one the number of constraints.

Appendix B. Data sets list

Table B.2: List of the 40 data sets used for experiments.

dataset	n	k	classes	dataset	n	k	classes
labor	57	11	2	ecoli	336	6	8
white_clover	63	6	4	liver_disorders	345	1	2
postoperative	90	8	3	ionosphere	351	33	2
zoo	101	16	7	monks3	554	6	2
lymph	148	18	4	monks1	556	6	2
iris	150	4	3	monks2	601	6	2
tae	151	2	3	credit_a	690	15	2
grub_damage	155	6	4	breast_w	699	9	2
hepatitis	155	16	2	diabetes	768	6	2
hayes_roth	160	3	3	anneal	898	31	6
wine	178	13	3	credit_g	1000	15	2
sonar	208	21	2	cmc	1473	9	3
glass	214	7	7	yeast	1484	7	10
heart_h	294	9	2	segment	2310	18	7
heart_c	303	11	2	kr_vs_kp	3196	36	2
haberman	306	2	2	hypothyroid	3772	25	4
solarflare_C	323	10	3	waveform	5000	19	3
solarflare_M	323	10	4	page_blocks	5473	10	5
solarflare_X	323	10	2	pendigits	10992	16	10
ecoli	336	6	8	nursery	12960	8	5