ECAI 2012
Luc De Raedt et al. (Eds.)

doi:10.3233/978-1-61499-098-7-264

Compression-based AODE Classifiers

G. Corani[1] and **A. Antonucci**[2] and **R. De Rosa** [3]

Abstract. We propose the COMP-AODE classifier, which adopts the compression-based approach [1] to average the posterior probabilities computed by different non-naive classifiers (SPODEs). COMP-AODE improves classification performance over the well-known AODE [10] model. COMP-AODE assumes a uniform prior over the SPODEs; we then develop the *credal* classifier COMP-AODE*, substituting the uniform prior by a *set* of priors. COMP-AODE* returns more classes when the classification is *prior-dependent*, namely if the most probable class varies with the prior adopted over the SPODEs. COMP-AODE* achieves higher classification utility than both COMP-AODE and AODE.

1 Introduction

Bayesian Model Averaging (BMA) weights the inferences produced by different candidate models, using as weights the posterior probabilities of the models themselves. This strategy is optimal if the set of candidate models includes the true one, as it is assumed by BMA; yet, this is not true in general and in fact BMA does *not* achieve good results in classification: see [2] and the references therein. The main problem is that BMA gets excessively concentrated around the single most probable model, as explained in [1]: "*averaging using the posterior probabilities to weight the models is almost the same as selecting the MAP model*", thus canceling the advantage of combining different models. The *compression-based* approach [1] overcomes this problem, making the weights less concentrated around a single model through a logarithmic smoothing of the models posterior probabilities. The compression-based weights can be justified from an information-theoretic viewpoint. In [1], the compression-based approach has been used 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 [10], 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. The posterior probabilities computed for the classes by the different SPODEs are then simply averaged. In [11] more sophisticated methods have been tested for aggregating SPODEs; yet, "*AODE, which simply linearly combines every SPODE without any selection or weighting, is actually more effective than the majority of rival schemes*". In particular, AODE outperforms BMA applied over SPODEs [11]; this is not surprising in the light of the previous discussion.

[1] Istituto Dalle Molle Intelligenza Artificiale (IDSIA), Switzerland, email: giorgio@idsia.ch
[2] Istituto Dalle Molle Intelligenza Artificiale (IDSIA), Switzerland, email: alessandro@idsia.ch
[3] Dip. di Scienze dell'Informazione, Universita' di Milano

The first contribution of this paper is the COMP-AODE classifier, which averages over the SPODEs using the compression-based coefficients. We present its algorithms and show, through extensive experiments, that it yields an improvement in classification performances over the standard AODE. However COMP-AODE, like most Bayesian ensembles of classifiers, adopts a uniform prior over the models in the attempt of being non-informative. Yet, the uniform prior represents a condition of *prior indifference* between the different models, while instead we generally are in a condition of *prior ignorance*. To effectively model prior ignorance we adopt the paradigm of *credal classification* (see [3] and the references therein), substituting the single uniform prior over the models by a (so-called *credal*) *set* of priors over the models, which represents prior ignorance by letting vary the prior probability of each model over a wide interval, instead of keeping it fixed to a specific number. We call COMP-AODE* the resulting classifier.

If the most probable class of an instance varies under different priors over the models, the classification is *prior-dependent*. Credal classifiers remains reliable on prior-dependent instances by returning a set of classes instead of a single class; the classification is in these cases *indeterminate*, namely more classes are returned. In Section 3 we show that on the prior-dependent instances, COMP-AODE* achieves high accuracy by returning a small set of classes. On the same instances, instead, COMP-AODE undergoes a severe drop of accuracy. Moreover, COMP-AODE* shows better empirical performances than previous 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 $P(A = a)$. We assume the data to be complete and the training data $\mathcal{D}$ to contain n instances. To learn the parameters of the SPODEs from the training data we adopt Bayesian estimation, using Dirichlet priors and setting the equivalent sample size to one. Under 0-1 loss, probabilistic classifiers return the *single* most probable class for each instance. Classifiers based on imprecise-probabilities (*credal* classifiers) change this paradigm, by instead returning more classes on the *prior-dependent* instances. We discuss this point more in detail in Section 2.3.

2.1 AODE

The AODE classifier [10] is based on a set $\mathcal{S} := \{s_1, \ldots, s_k\}$ of k SPODEs (SuperParent-One-Dependence Estimator); in particular, SPODE s_j has A_j as super-parent, namely it models all the remaining features as depending on both A_j and on the class C, as shown in Figure 1.

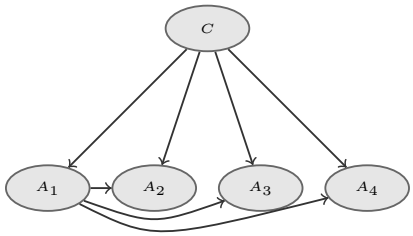

Figure 1. SPODE s_1 with super-parent A_1.

Such a topology induces in the joint probabilities of the SPODE s_j the following factorization:

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

To classify a test instance $\tilde{\mathbf{a}}$, AODE simply averages the posterior probability $P(c|\tilde{\mathbf{a}})$ computed by each SPODE. In this paper we study alternative approaches to aggregate the predictions of the SPODEs.

2.2 COMP-AODE: compression-based AODE

Compression-based averaging [1] overcomes the problem of BMA getting excessively concentrated around the single most probable model by replacing the posterior probabilities of the models with smoother, *compressed*, weights. We denote by $\eta(s_j|\mathcal{D})$ the weight assigned to model s_j.

We introduce a *null classifier*, denoted by s_0, 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 is necessary to compute the compression coefficients.

Considering that all the SPODEs have the same number of variables, the same number of arcs and the same maximum in-degree[4], we assign the same prior probability to each SPODE. We instead assign prior probability ϵ=0.01 to the null model.Thus, we can consider a variable S with values in $\mathcal{S} \cup \{s_0\}$, and define the following prior mass function:

$$P(s_j) = \begin{cases} \epsilon & j = 0, \\ \frac{1-\epsilon}{k} & j = 1, \ldots, k. \end{cases} \quad (2)$$

We compute the *conditional* log-likelihood of SPODE s_j as in [1]:

$$LL_j := \sum_{i=1}^{n} \log(P(c^{(i)}|\mathbf{a}^{(i)}, s_j, \hat{\theta}_j)), \quad (3)$$

where θ_j is a variable over the set of parameters for the model, and $\hat{\theta}_j$ is the value of its Bayesian estimation. It could be also possible to evaluate the probability of the data given SPODE s_j by the *marginal* likelihood, i.e., $P(\mathcal{D}|s_j) = \int P(\mathcal{D}|s_j, \theta_j)P(\theta_j|s_j)d\theta_j$. Yet, the marginal likelihood measures how good the model is at representing the *joint* distribution, while a classifier has instead to estimate the posterior probability of the classes by *conditioning* on the value of the features. Therefore, a model can poorly perform in classification despite a high marginal likelihood [4]; the conditional likelihood is a more appropriate score for classifiers.

[4] The in-degree is the number of parents of a node, and its maximum value is two for any SPODE.

If LL_0 is the conditional log-likelihood of the null model, then $LL_0 := -nH(C)$, where $H(C) := -\sum_{c\in\mathcal{C}} P(c)\log P(c)$ is the *entropy* of the class[5] [1]. The compression coefficients are computed in two steps: computation of the *raw* coefficients and normalization. For $j \neq 0$, the *raw* compression coefficient of SPODE s_j is defined as:

$$\tilde{\eta}_j := 1 - \frac{\log P(s_j|\mathcal{D})}{\log P(s_0|\mathcal{D})} = 1 - \frac{LL_j + \log\frac{1-\epsilon}{k}}{-nH(C) + \log\epsilon}. \quad (4)$$

If $\tilde{\eta}_j$ is negative, s_j is a worse predictor than the null model; if it is positive, as it normally happens, s_j performs better than the null model. The upper limit of $\tilde{\eta}_j$ is 1, in which case s_j is a perfect predictor. Following [1], we keep in the ensemble the *feasible* models with $\tilde{\eta}_j > 0$, and we remove from the ensemble the remaining ones. This corresponds to removing from the ensemble the models whose posterior probability falls below a certain threshold, which is sometimes done also when computing BMA. Since by definition $\tilde{\eta}_0 = 0$, the null model is not part of the resulting ensemble.

Using the compression coefficients can be justified as follows [1]: $LL_j + \log P(s_j)$ "*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 potentially 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.*"

Without loss of generality, we assume the features to be ordered, so that features $\{A_1, A_2, \ldots, A_{\tilde{k}}\}$ yield a feasible model when used as super-parent; i.e., SPODEs $\{s_1, s_2, \ldots, s_{\tilde{k}}\}$ are the feasible ones. The *normalized* compression coefficients, which we denote as $\eta(s_j|\mathcal{D})$, are obtained by normalizing the raw compression coefficients of the feasible SPODEs:

$$\eta(s_j|\mathcal{D}) = \begin{cases} \frac{\tilde{\eta}_j}{\sum_{l=1}^{\tilde{k}} \tilde{\eta}_l} & \text{if} \quad j = 1, \ldots, \tilde{k}, \\ 0 & \text{otherwise.} \end{cases} \quad (5)$$

The posterior probabilities computed by COMP-AODE for instance $\tilde{a}$ ares:

$$P(c|\tilde{\mathbf{a}}) = \sum_{j=1}^{k} P(c|\tilde{\mathbf{a}}, s_j) \cdot \eta(s_j|\mathcal{D}). \quad (6)$$

2.3 Introducing a Set of Priors: COMP-AODE*

We extend COMP-AODE to *imprecise probabilities* [9], by allowing multiple specifications of the prior mass function $P(S)$; we denote by $K(S)$ the so-called *credal set* of prior mass functions. A uniform mass function represents prior *indifference* between the different SPODEs; instead, a credal set provides a more cautious model of prior *ignorance* about which SPODE might have produced the data.

In principle we could let the prior probabilities of each SPODE vary exactly between zero and one by considering any possible prior (*vacuous* model). Yet, this prevents learning from data, generating vacuous posterior inferences. To obtain non-vacuous posterior inferences, we introduce a non-zero lower bounds in the credal set $K(S)$, which is defined as follows:

$$K(S) := \left\{ P(S) \;\middle|\; \begin{array}{l} P(s_0) = \epsilon \\ P(s_j) \geq \epsilon \quad j = 1, \ldots, k \\ \sum_{j=0}^{k} P(s_j) = 1 \end{array} \right\} \quad (7)$$

[5] For this equivalence to hold, we compute the entropy using the natural logarithm, instead of the log in basis 2.

The set in (7) is convex; its k extreme distributions are those assigning mass ϵ to all the models apart from a single SPODE, which get mass $1 - k\epsilon$. When $P(S)$ varies in $K(S)$, the *raw* coefficients of compression, defined as in (4), span the following range:

$$\eta_j \in \left[1 - \frac{LL_j + \log \epsilon}{-nH(C) + \log \epsilon},\ 1 - \frac{LL_j + \log(1 - k\epsilon)}{-nH(C) + \log \epsilon}\right]. \quad (8)$$

However, the different η_j cannot vary in the intervals of Eq.(8) independently from each other, because of the normalization constraint of Eq. (7). Note moreover that since the prior of Eq.(2) is contained in the credal set, the point estimate of the compression coefficient used by COMP-AODE belongs to the interval.

We regard SPODE s_j as non-feasible if the *upper* bound of the interval (8) is non-positive; this approach is particularly conservative as it preserves the models which are feasible (in the sense of Section 2.2) for at least a prior in the set $K(X)$. COMP-AODE* is thus more conservative than COMP-AODE, namely it removes from the ensemble a lower number of models. However, in practice, no SPODE is generally removed from the ensemble neither by COMP-AODE*, nor by COMP-AODE.

Considering that $K(S)$ is a set a prior mass functions, COMP-AODE* can be interpreted as a set of COMP-AODE classifiers, each induced in correspondence of a different prior. An instance is *prior-dependent* if the most probable class varies under the different priors of the credal set. COMP-AODE* remains reliable on prior-dependent instances, by returning a set of classes instead of a single one. Returning a set of classes on the *prior-dependent* instances and a single class on the remaining *safe* instances is in fact the typical behavior of credal classifiers [3]. Note that prior-dependence is not a characteristic of the instance alone: a credal-classifier might judge an instance as prior-dependent, and an alternative credal classifier might judge it as safe. Thus, prior-dependence is a characteristic of a certain instance, when analyzed by a *specific* credal classifier.

2.3.1 Credal Dominance

Without loss of generality we assume the features reordered, so that the first $\tilde{k}$ features yield a feasible model, i.e., SPODEs $\{s_1, \ldots, s_{\tilde{k}}\}$ are the feasible ones. Given an unsupervised test instance $\mathbf{a}$ whose class is unknown, class $c' \in \mathcal{C}$ dominates $c'' \in \mathcal{C}$ if c' is more probable than c'' under any prior of the credal set, i.e. if:

$$\min_{P(S) \in K(S)} \frac{P(c'|\mathbf{a})}{P(c''|\mathbf{a})} = \frac{\sum_{j=1}^{\tilde{k}} P(c'|\mathbf{a}, s_j)\tilde{\eta}(s_j|\mathcal{D})}{\sum_{i=1}^{\tilde{k}} P(c''|\mathbf{a}, s_i)\tilde{\eta}(s_i|\mathcal{D})} > 1, \quad (9)$$

where, as we noted before, the sum over the feasible models corresponds to that over the first $\tilde{k}$ models and where we have substituted the standardized compression coefficients by the raw ones, having considered that $\sum_{j=1}^{\tilde{k}} \tilde{\eta}(s_j|\mathcal{D})$ is positive by definition. The function to be minimized in (9) then becomes:

$$\frac{\sum_{j=1}^{\tilde{k}} P(c'|\mathbf{a}, s_j)\left(\log \epsilon - nH(C) - LL_j - \log P(s_j)\right)}{\sum_{i=1}^{\tilde{k}} P(c''|\mathbf{a}, s_i)\left(\log \epsilon - nH(C) - LL_i - \log P(s_i)\right)}.$$

By setting, for each $j = 1, \ldots, \tilde{k}$, $x_j := \log P(s_j)$, $\alpha_j := P(c'|\mathbf{a}, s_j)$, $\beta_j := P(c''|\mathbf{a}, s_j)$, and

$$\begin{bmatrix} \delta \\ \gamma \end{bmatrix} := -\sum_{j=1}^{\tilde{k}} \begin{bmatrix} P(c'|\mathbf{a}, s_j) \\ P(c''|\mathbf{a}, s_j) \end{bmatrix} \left(\log \epsilon - nH(C) - LL_j\right). \quad (10)$$

the optimization problem to check whether c' dominates c'' becomes:

$$\begin{aligned} \min_{x_1, \ldots, x_{\tilde{k}}} \quad & \frac{\sum_{j=1}^{\tilde{k}} \alpha_j x_j + \delta}{\sum_{j=1}^{\tilde{k}} \beta_j x_j + \gamma}, \\ \text{subject to} \quad & \\ & x_j \geq \log \epsilon \quad j = 1, \ldots, \tilde{k}, \\ & \textstyle\sum_{j=1}^{\tilde{k}} e^{x_j} = 1 - \epsilon - (k - \tilde{k})\epsilon. \end{aligned}$$

The last constraint is related to the normalization constraint in the definition (7) of the credal set, and imposes the sum of the prior probability of the feasible SPODEs to be one minus the prior probabilities of the $k - \tilde{k}$ non-feasible SPODEs and the null model, whose prior probability is set to ϵ. We then substitute $y_j := e^{x_j}$ to avoid numerical problems in the optimization, thus getting a non-linear optimization problem with linear constraints.

A class is *non-dominated* if no alternative class dominates it according to the test of Eq.(9). COMP-AODE* identifies the set of non-dominated classes through the *maximality* approach [9], which is commonly adopted for decision making with imprecise probabilities; for each instance it requires running the dominance test on each pair of classes, as formalized by Algorithm 1.

Since the credal set (7) includes the prior adopted by COMP-AODE, the non-dominated classes returned by COMP-AODE* include by design the most probable class identified by COMP-AODE; thus, when COMP-AODE* returns a single class, it is the same class returned by COMP-AODE.

Algorithm 1 Identification of the non-dominated classes $\mathcal{ND}$ through maximality

```
ND := C
for c' ∈ C do
  for c'' ∈ C (c' ≠ c'') do
    compute the dominance test of Eq.(9)
    if c' dominates c'' then
      remove c'' from ND
    end if
  end for
end for
return ND
```

2.4 Complexity

To analyze the computational complexity of the classifiers, we distinguish between the *learning* and the *classification complexity*, the latter referring to the classification of a single instance. We analyze both the *space* and the *time* required for computations. The orders of magnitude 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 given in Table 1.

A single SPODE s_j requires storing the tables $P(C)$, $P(A_j|C)$ and $P(A_i|C, A_j)$, with $i = 1, \ldots, k$ and $i \neq j$, implying space complexity $\mathcal{O}(lkv^2)$ for learning each SPODE and $\mathcal{O}(lk^2v^2)$ for the AODE ensemble. For each classifier, the same tables should be available during learning and classification; 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 as in Eq.(1) is $\mathcal{O}(lk)$ for each SPODE, and hence

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

Table 1. Complexity of classifiers.

$\mathcal{O}(lk^2)$ for AODE. Learning COMP-AODE takes roughly the same space as AODE, but higher computational time, due to the evaluation of the conditional likelihood of Eq.(3). The additional computational time is $\mathcal{O}(nlk)$, thus requiring $\mathcal{O}(n(l+k)k)$ time overall. For classification, time and space complexity is equivalent to that of AODE.

COMP-AODE* has the same space complexity of COMP-AODE and the same time complexity in learning, but higher time complexity in classification. The pairwise dominance tests in Algorithm 1 require solving a number of optimization problems for each test instance which is quadratic in the number of classes. Each optimization has time complexity which is roughly cubic in the number of constraints/variables, which is in turn $\mathcal{O}(k)$.

Compared to AODE, the new classifiers require higher training time, while the higher cautiousness characterizing COMP-AODE* increases by one the exponents of the number of classes and attributes in the complexity of the classification time.

3 Experiments

We run experiments on 40 UCI data sets, taken from the UCI repository; the sample size ranges between 57 (labor) and 12960 (nursery); the number of classes between 2 and 10 (pendigits). On each data set we perform 10 runs of 5-folds cross-validation. Missing data are replaced by the median/mode for numerical/categorical features, so that all data sets are complete. We discretize numerical features by the MDL-based discretization [6]. For AODE, we set to 1 the frequency limit; namely, features with a frequency in the training set below this value are not used as parents; this is also the default value in WEKA.

In order to compare two classifiers over the collection of data sets we use the non-parametric Wilcoxon signed-rank test.[6] This test is indeed recommended for comparing two classifiers on multiple data sets [5]: being non-parametric it both avoids strong assumptions and deals robustly with outliers.

3.1 AODE vs. COMP-AODE

We consider 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 and $P(c^{(i)}|\mathbf{a}^{(i)})$ is the probability estimated by the classifier for the true class of the i-th instance of the test set. In Figure 2(a) we show the *relative accuracies*, namely the accuracy of COMP-AODE divided, separately for each data set, by the accuracy of AODE. Thus, better performance relatively to AODE is achieved when the relative accuracy is >1. The accuracy of the two models is identical (relative accuracy = 1) in 15/40 cases ; in 14/40 data sets relative accuracy is >1 (COMP-AODE wins) while in 11/40 data sets relative accuracy is <1 (AODE wins). Overall, the performance of COMP-AODE and AODE on this collection of data sets is not significantly different. In Fig.2(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; thus, better performance relatively to AODE is achieved when the relative Brier loss is <1. The Brier loss is more sensitive than accuracy, and thus magnifies the differences among classifiers, as can be seen by comparing the scales of relative accuracies and relative Brier losses. Under Brier loss, COMP-AODE performs significantly better than AODE (p-value<0.01). The Brier loss of the two models is identical (relative loss = 1) in 12/40 cases ; in 23/40 data sets relative loss is <1 (COMP-AODE wins) while only in 5/40 data sets relative loss is >1 (AODE wins). This is noteworthy given the high performance of AODE and the fact the standard AODE often outperforms alternative weighting methods over SPODEs [11]. Our findings thus extend the results of [1] in which the compression-based approach was successfully applied over an ensemble of naive Bayes classifiers.

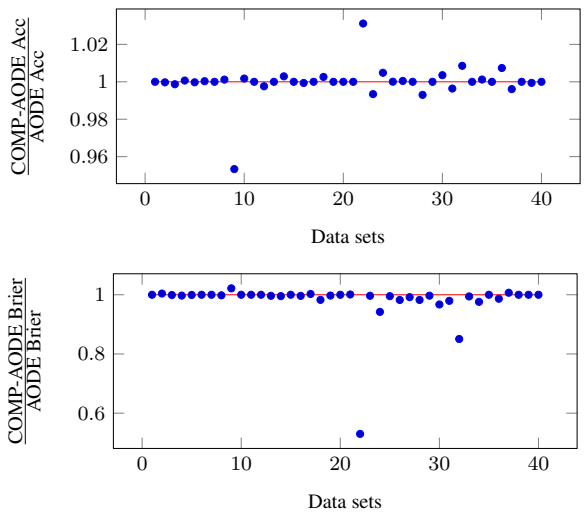

Figure 2. Relative accuracies and relative Brier losses: for accuracy, performance better than AODE corresponds to points lying *above* the horizontal line; for Brier loss, performance better than AODE corresponds to points lying *below* the horizontal line. Note the different scale of the two graphs, reflecting the higher sensitivity of Brier loss.

3.2 Evaluation of COMP-AODE*

A credal classifier separates in fact the instances into two groups: the *safe* ones, for which a single class is returned, and the *prior-dependent* ones, for which instead different non-dominated classes are returned. To fully characterize the performance of an imprecise classifier, four indicators can be considered: *determinacy*: the proportion of instances recognized as safe and thus classified with a single class; *single-accuracy*: the accuracy achieved over the instances recognized as safe; *set-accuracy*: the accuracy achieved over the prior-dependent instances, by returning a set of classes; *indeterminate output size*: the average number of classes returned on the prior-dependent instances.

COMP-AODE* is generally very determinate: its average determinacy is 99%; this means that on average it recognizes only 1% of the instances as prior-dependent. This is probably a consequence of

[6] We use the test as follows: for a given indicator we build two *paired* vectors, one for each classifier: the same position refers, in both vectors, to the same data set. The two vectors are then used as input for the test.

the logarithmic smoothing induced by the compression coefficients, which makes the weights of the models little sensitive on the chosen prior. We see this robustness to the choice of the prior as a desirable and previously unknown property of the compression-based approach; in fact, it is easy investigating this point only once developed the credal classifier. The robustness to the choice of the prior might well constitute a further reason for the good empirical performance of the compression-based approach. COMP-AODE* performs well when indeterminate: averaging over all data sets, it achieves 95% set-accuracy by returning 2 classes. In Fig.3, we compare the accuracy achieved by COMP-AODE on the instances judged respectively safe and prior-dependent by COMP-AODE*. Each point refers to a different data set; for that data set, it represents the accuracy achieved by COMP-AODE on the safe instances (y -coordinate) and on the instances judged as prior-dependent by COMP-AODE* (x -coordinate). On almost every data set, the accuracy of COMP-AODE is much higher on the safe instances than on the prior-dependent instances ($y \gg x$) ; the drop of accuracy between the safe and the prior-dependent instances is indeed significant (p-value < 0.01). As a rough indication, averaging over data sets, the accuracy of COMP-AODE is 82% on the safe instances but only 47% on the prior-dependent instances. Thus, while COMP-AODE provides fragile classifications on the prior-dependent instances, COMP-AODE* remains reliable by returning a small-sized but highly reliable set of classes. Thus 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 these instances, as already discussed, COMP-AODE* preserves its reliability thanks to indeterminate classifications.

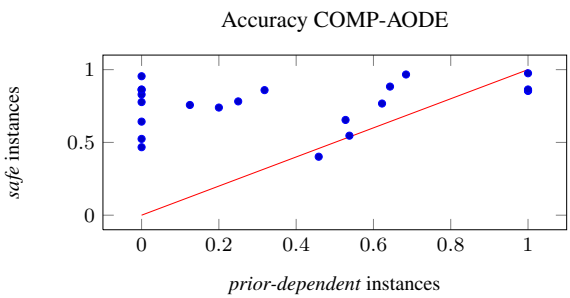

Figure 3. Accuracy of COMP-AODE on the instances recognized as safe and as prior-dependent by COMP-AODE*; the straight line is the *bisectrix*.

3.3 Utility-based Measures

We have seen that COMP-AODE* extends in a sensible way COMP-AODE, being able to recognize prior-dependent instances and to robustly deal with them. To further compare COMP-AODE and COMP-AODE*, we adopt the utility-based measures of [12]. In fact, how to compare determinate and indeterminate predictions is far from obvious. The *discounted accuracy* rewards a prediction made of m classes with $1/m$ if it contains the actual class, and 0 otherwise; the discounted accuracy of a credal classifier can be then compared to the accuracy of a determinate classifier. However [12] points out some severe limits of discounted-accuracy, which we illustrate by means of an example. We consider two medical doctors, *random* and doctor *vacuous*, whose task is to classify each patient in one of the two categories {healthy, diseased}. Doctor random generates random diagnoses, drawing its judgment from a uniform probability mass function. Doctor vacuous instead always returns both categories, admitting to be ignorant. Let us assume that the hospital receives a quantity of money which is proportional to the discounted-accuracy generated by its doctors when visiting patients. Both doctors provide the same *expected* discounted-accuracy (1/2) and thus the same *expected* profit; yet, the profit generated by doctor vacuous is deterministic, while the profit generated by doctor random is affected by considerable variance. Under any risk-averse utility function, doctor vacuous generates a higher *utility* than doctor random, yielding the same expected reward but with less variance: under risk-aversion, the expected utility increases with expectation of the rewards and decreases with their variance (see the references in [8]). In [12] it is thus proposed to compare credal and determinate classifiers measuring the *utility* of the reward constituted by discounted-accuracy: the stronger the risk aversion, the higher the value of indeterminate but accurate (containing the true class in the set of non-dominated classes) predictions. In [12] the utility of a correct and determinate classification (discounted-accuracy 1) is set to 1; the utility of a wrong classification (discounted-accuracy 0) to 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, depending on the degree of aversion. In correspondence of these two values, two quadratic utility functions are derived: u_{65} passes through $\{u(0) = 0, u(0.5) = 0.65, u(1) = 1\}$, while u_{80} passes through $\{u(0) = 0, u(0.5) = 0.8, u(1) = 1\}$. In real applications the utility function should 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. Since $u(1) = 1$, the utility and the accuracy of a traditional classifier coincide; therefore the utility values of credal classifiers can be directly compared with the predictive accuracy of the traditional classifiers. In [7] 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 these utility functions.

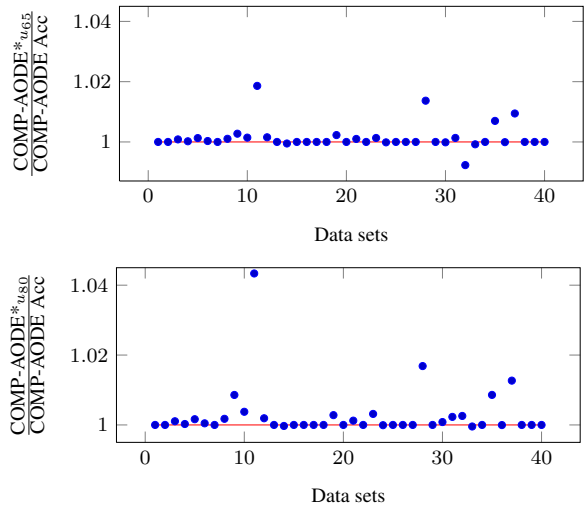

Figure 4. Relative utilities; a better performance of COMP-AODE* over COMP-AODE is represented by points lying *above* the horizontal line.

Figures 4(a) and 4(b) show the the utility of COMP-AODE* di-

vided, data set by data set, by the accuracy of COMP-AODE. The two plots refer respectively to u_{65} and u_{80}. Some points are exactly 1, since in some data sets COMP-AODE is completely determinate. However, the points tend to be generally higher than 1; COMP-AODE* generates significantly higher utility (p-value < 0.01) than COMP-AODE under both u_{65} and u_{80}. The numerical improvement is generally small, being close to 1%; however this is reasonable if we consider that COMP-AODE* has 99% determinacy on average. The improvement of COMP-AODE* over COMP-AODE is more evident under u_{80}, due to the higher utility associated in this case to classifications which are accurate but indeterminate. Moreover, 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 improved 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 Other Credal Classifiers

Previous credal classifiers, which return more classes on the instances identified as prior-dependent, include for instance the *naive credal classifier* (NCC), namely an extension of naive Bayes to imprecise probability and the *credal model averaging* (CMA), a generalization of BMA over naive Bayes classifiers, in the same spirit of Section 2.3 but without compression. We point the reader to [3] for more insights and references on previous credal classifiers. Here we compare the performances of NCC, CMA and COMP-AODE* by means of the utility measures u_{65} and u_{80}, adopting the same experimental setup detailed at the begin of Section 3. We compare these classifiers over the collection of 40 data sets by the Friedman test coupled with the Nemenji post-hoc, as recommended in [5]. Under both u_{65} and u_{80} we thus eventually rank the credal classifiers according to the utility they generate. Figure 5 reports the results of this comparison for both u_{65} and u_{80}. The post-hoc analysis, under u_{65}, ranks COMP-AODE* as significantly better than both CMA and NCC. Under u_{80} no significant difference is found among classifiers; the point is that both CMA and NCC are much more indeterminate than COMP-AODE*, and benefit at a much larger extent than COMP-AODE* from the increase utility assigned by u_{80} to indeterminate but accurate classifications; in this way, they close the gap with COMP-AODE*. However, also under u_{80} COMP-AODE* has the highest average rank, and we conclude that COMP-AODE* provides a generally higher classification performance than both CMA and NCC.

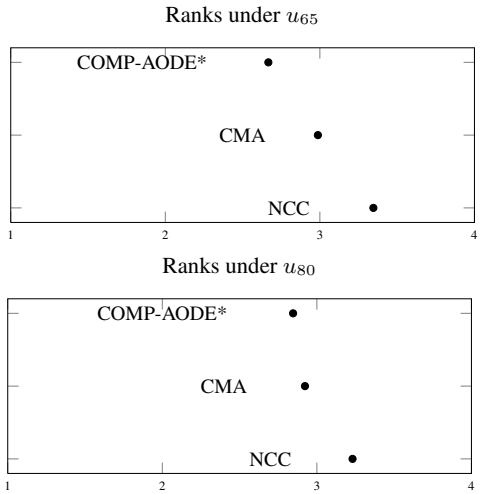

Figure 5. Comparison between credal classifiers. The points denote the average ranks, while the bars display the critical distance. The average ranks of two classifiers are significantly different if they differ by more than the critical distance, namely if their bars do not overlap.

4 Conclusions

COMP-AODE is a new classifier based on compression-based averaging of SPODEs; it slightly but significantly improves classification performance over AODE. COMP-AODE* extends it to imprecise probability, by replacing the single uniform prior over SPODEs with a credal set of priors. COMP-AODE* returns more classes on the instances recognized as prior-dependent and achieves higher prediction utility than both COMP-AODE and AODE.

Acknowledgments

Research partially supported by Swiss NSF grant no. 200020-132252 and the Hasler foundation grant n. 10030.

References

[1] M. Boullé, 'Compression-based averaging of selective naive Bayes classifiers', *Journal of Machine Learning Research*, **8**, 1659–1685, (2007).
[2] J. Cerquides, R.L. De Màntaras, et al., 'Robust Bayesian linear classifier ensembles', *Lecture notes in computer science*, **3720**, 72, (2005).
[3] G. Corani, A. Antonucci, and M. Zaffalon, 'Bayesian networks with imprecise probabilities: Theory and application to classification', in *Data Mining: Foundations and Intelligent Paradigms*, eds., D. E. Holmes, L. C. Jain, and J. Kacprzyk, volume 23, 49–93, Springer, (2012).
[4] R. Cowell, 'On searching for optimal classifiers among Bayesian networks', in *Proceedings of the Eighth International Conference on Artificial Intelligence and Statistics*, pp. 175–180, (2001).
[5] J. Demsar, 'Statistical comparisons of classifiers over multiple data sets', *Journal of Machine Learning Research*, **7**, 1–30, (2006).
[6] U. M. Fayyad and K. B. Irani, 'Multi-interval discretization of continuous-valued attributes for classification learning', in *Proc. 13th Int. Joint conference on artificial intelligence (IJCAI-93)*, pp. 1022–1027, (1993).
[7] J. Jose del Coz and A. Bahamonde, 'Learning nondeterministic classifiers', *Journal of Machine Learning Research*, **10**, 2273–2293, (2009).
[8] H. Levy and H.M. Markowitz, 'Approximating expected utility by a function of mean and variance', *The American Economic Review*, **69**(3), 308–317, (1979).
[9] P. Walley, *Statistical Reasoning with Imprecise Probabilities*, Chapman and Hall, New York, 1991.
[10] G.I. Webb, J.R. Boughton, and Z. Wang, 'Not so naive Bayes: Aggregating one-dependence estimators', *Machine Learning*, **58**(1), 5–24, (2005).
[11] Y. Yang, G.I. Webb, J. Cerquides, K.B. Korb, J. Boughton, and K.M. Ting, '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, (2007).
[12] M. Zaffalon, Corani G., and D. Mauá, 'Utility-based accuracy measures to empirically evaluate credal classifiers', in *ISIPTA'11: Proceedings of the Seventh International Symposium on Imprecise Probability: Theories and Applications*, pp. 401–410, (2011).