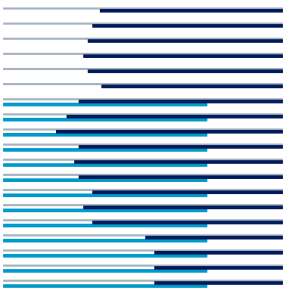
Evaluating credal classifiers by utility-discounted predictive accuracy

Marco Zaffalon Giorgio Corani Denis Mauá



Technical Report No. IDSIA-03-12
February 2012

IDSIA / USI-SUPSI
Dalle Molle Institute for Artificial Intelligence
Galleria 2, 6928 Manno, Switzerland

Evaluating credal classifiers by utility-discounted predictive accuracy

Marco Zaffalon Giorgio Corani Denis Mauá

February 2012

Abstract

Predictions made by imprecise-probability models are often indeterminate (that is, set-valued). Measuring the quality of an indeterminate prediction by a single number is important to fairly compare different models, but a principled approach to this problem is currently missing. In this paper we derive, from a set of assumptions, a metric to evaluate the predictions of credal classifiers. These are supervised learning models that issue set-valued predictions. The metric turns out to be made of an objective component, and another that is related to the decision-maker's degree of risk aversion to the variability of predictions. We discuss when the measure can be rendered independent of such a degree, and provide insights as to how the comparison of classifiers based on the new measure changes with the number of predictions to be made. Finally, we make extensive empirical tests of credal, as well as precise, classifiers by using the new metric. This shows the practical usefulness of the metric, while yielding a first insightful and extensive comparison of credal classifiers.

1 Introduction

When we use an imprecise-probability model to make predictions, we meet one of the most striking differences of imprecise probability in comparison to precise probability: the imprecise-probability model can issue *indeterminate* predictions. That is, among the set of possible options, the model may drop some of them as sub-optimal, while keeping the entire remaining set as its prediction. The prediction is generally indeterminate as such a set is not necessarily a singleton. Indeterminate predictions are a crucially important feature of imprecise-probability models: they allow credible, and reliable, predictions to be obtained no matter how scarce is the information available to build a model.

Yet, we should have a way to *measure* how good is an indeterminate prediction. A major reason is that we need to compare imprecise- with precise-probability models: we should have a simple, and possibly shared way to say which one is better, in a given application. The same consideration applies when we compare two imprecise-probability models. Ideally, we would like to be able to reward each prediction by a single number, be it determinate or indeterminate . Most probably this would speed up progress in the field, as it would enable comparisons to be automatized over a large number of test applications.

In the case of precise-probability models, there are well-consolidated measures to do this. Let us consider the field of *pattern classification* [9], which is the focus of this paper (Section 2 gives a brief introduction to classification problems). In this case, the predictive models are called (precise) *classifiers*. A classifier predicts one out of a finite set $\mathscr{C}$ of so-called *classes*. In this case, correct predictions may be rewarded with 1 and incorrect ones with 0, thus giving rise to the measure of performance called the *predictive accuracy* of a classifier: i.e., the proportion of correct predictions it makes.

The situation is very different with *credal classifiers*, that is, classifiers that issue set-valued predictions. One of the few proposals to evaluate an indeterminate prediction by a single number can be found in [5]: a prediction made of a set $\mathscr{K}$ of k classes is rewarded with $1/k$ if it contains the actual class, and with 0

otherwise. This gives rise to the measure called *discounted accuracy*, which was borrowed from the field of multi-label classification [18]. The problem here is that no justification is given for discounted accuracy, as the work in [5] points out. In [13], classifiers which return indeterminate classifications are evaluated through the F-metric, originally designed for information retrieval problems; but also in this case, there is no actual justification for the choice of this measure. Other than these, the past proposals are either explicitly non-numerical, as the rank test in [5], or require a vector of parameters to evaluate the performance, as in [4]. The latter approach is meaningful, but was conceived to compare credal with precise classifiers, and cannot be easily generalized to the more general case; moreover, it is a method that needs supervision so that it does not easily lend itself to be run automatically on many test cases.

In our view, the scarcity of principled numerical evaluation methods for credal classifiers is not accidental: in fact, it is not easy to assign a single number to an indeterminate prediction. Consider the following case: there is a *vacuous* classifier, which every time predicts the set of all classes $\mathscr{C}$, and a *random* one, which picks up a class from $\mathscr{C}$ through the uniform distribution. If $\mathscr{C}$ is made of two classes (we say that the classification problem is binary), and we use the predictive accuracy, the random classifier has an expected reward equal to $1/2$. What should be the expected reward of the vacuous classifier? Both classifiers do not know how to predict the class, but only the vacuous classifier declares it. From this, one might argue that the latter should be rewarded with more than $1/2$. On the other hand, it is clear also that the vacuous classifier cannot predict the class better than the random one, so that one might argue that it should be rewarded with $1/2$ too.

In the attempt to address these kinds of problems in the most objective way, we found it useful to regard classifiers as bettors. In the betting framework introduced in Section 3, we assume we only know how to value determinate predictions, in particular by 0-1 rewards. In Section 4, we extend the framework, in a kind of least-committal way, to credal classifiers: we show that, under reasonable assumptions, indeterminate predictions should be valued according to discounted accuracy.

Note, however, that discounted accuracy values the vacuous and the random classifiers the same. This kind of (questionable) effect can be traced back to having deliberately avoided introducing subjective considerations in the evaluation. Still, subjective preferences should be accounted for: we introduce in Section 5 a decision maker in charge of selecting the 'best' classifier in the next bet, and show that preferences can enter the picture through his utility, as a function of discounted accuracy. This defines the measure we propose to evaluate credal classifiers: *utility-discounted predictive accuracy*. More generally speaking, this shows in a very definite sense how the reliability of a classifier is tightly related to the variability of its predictions, and that the aversion to this variability is what makes some people prefer credal classifier to precise ones. In Section 6 we briefly discuss how utility-discounted accuracy can equivalently, and quite naturally, be derived also in a framework based on money bets. This illustrates the tight relationship of our approach with finance.

In Section 7 we discuss an important case where the evaluation can still be made in quite an objective way despite the decision-maker's subjective preferences, and we relate this to the amount of indeterminacy produced by a credal classifier. In Section 8 we analyze how the picture changes if we focus on evaluating classifiers in the next $m \geq 1$ bets. We show that the difference between precise and credal classifiers decreases with growing m, so that the relative benefits of credal classification are less pronounced for large m.

Sections 9 is entirely devoted to the empirical evaluation of credal classifiers based on utility-discounted accuracy. We start by discussing the choice of sensible utility functions for our aim, that is, the fair comparison of credal classifiers.

In Section 9.1, we focus the comparison on two classifiers representative of the precise and the credal classes: *naive Bayes classifier* (NBC [7]) and *naive credal classifier* (NCC [20, 21, 4]). We infer and test them both on artificial data and on 55 real data sets from the well-known UCI repository. This shows, as expected, that NCC yields particular advantages over NBC on small learning sets. In more general conditions, the situation is either slightly or decidedly in favor of NCC, depending on the preference of the decision maker toward reliability of classification. It is shown in particular that when the NCC is indeterminate, the NBC issues fragile predictions; this confirms past results in the literature.

The comparison is extended in Section 9.2 to other credal classifiers, again on the UCI data sets. A thorough analysis highlights different characterists of the involved credal classifiers, and, at the same time, a substantial balance in their predictive performances (in particular when they are used jointly with feature selection to reduce overfitting). It is worth remarking that this kind of extensive and insightful comparison is made here for the first time, thanks to the availability of utility-discounted accuracy.

Section 9.3 is devoted to a detailed comparison of NCC with a set-valued classifier developed within precise probability: the *non-deterministic* classifier (NDC) of Coz et al. [13]. This comparison is interesting because the mentioned classifiers work according to very different principles, even though both yield set-valued classifications. Moreover, the discussion shows how the F-metric used in [13] for comparing classifiers, can be meaningfully re-interpreted as a utility function in our framework. This provided a justification for such a metric that was not available before. The comparison of NCC with NDC gives us the opportunity to consider a related question in Section 9.4: the extent to which a fair comparison of classifiers is possible to do when some of them implement the so-called *rejection option*: this is a precise-probability tool to issue set-valued classifications that relies on the inspection of the posterior probabilities assigned to the classes.

After our concluding remarks in Section 10, we provide the detailed numerical comparisons of classifiers, data set by data set, in Appendix A.

2 Classification problems

A classification problem is made of objects described by *attribute* (or *feature*) variables, which we group into the single variable A, and a class variable C. The class variable represents the object's category. There are finitely many possible categories, which we identify with their indexes to simplify notation: $\{1,\ldots,n\} =: \mathscr{C}$. We denote by c the generic element of $\mathscr{C}$. The attribute variable represents some characteristics of the object that are related to the class. Variable A takes values in the set $\mathscr{A}$; we denote by a its generic element. As an example, objects might be patients; A would represent information about a patient, such as personal information as well as outcomes of medical tests; $\mathscr{C}$ would index the patient's possible diseases.

Usually, some values of (A,C) are sampled in an independent and identically distributed way according to a law that is not known a priori. The so-called *learning set* $\mathscr{L}$ records those values, which are also called *instances* of (A,C). The goal of classification is to learn from the learning set a function that maps attributes into classes. We call this function a (*precise*) *classifier*.

A classifier is applied to predict the class of new objects based on their attributes. Predictions are rewarded through a *reward matrix* $\mathbb{R}$. This is an $n \times n$ matrix whose generic element r_{ij} is a number representing the reward obtained by predicting class i when the actual class is j. Equivalently, we can regard the reward matrix as a set of *gambles* (i.e., bounded random variables) $\mathbb{R}_i$, $i = 1,\ldots,n$, each one corresponding to a row of $\mathbb{R}$: gamble $\mathbb{R}_i$ represents the uncertain reward obtained by predicting class i and is defined by $\mathbb{R}_i(j) := r_{ij}$, with $j \in \mathscr{C}$. The reward matrix is an input of the classification problem, in the sense that it is given.

In classification, at least with respect to the machine learning practice, rewards are usually measured in a linear utility scale: although this point is often left implicit, we can deduce it from the observation that the performance of a classifier is usually identified with its expected reward.

The most frequent practice consists also in using just a 0-1 valued reward matrix, which we denote by $\mathbb{I}$ (Table 1 shows matrix $\mathbb{I}$ for the case of a binary classification problem). In this case, the gamble corresponding to the i-th row of the matrix coincides with the indicator function of set $\{i\}$, which yields $\mathbb{I}_i(i) = 1$, and $\mathbb{I}_i(j) = 0$ for $i \neq j$. Accordingly, the performance of a classifier corresponds to the probability of predicting the actual class. Such a probability is called the *predictive accuracy* (or simply the *accuracy*) of a classifier.

The term 'accuracy' is used also for the sample estimate of such a probability. In fact, a classification problem usually comes with a test set $\mathscr{T}$. This set contains a number of sampled instances of (A,C) that are

		Actual class	
		1	2
Predicted class	1	1	0
	2	0	1

Table 1: 0-1 reward matrix for a binary classification problem.

used to evaluate the classifier's predictive performance by measuring its accuracy on them. And in fact the predictive accuracy is by far the most frequently used empirical index to compare classifiers, even though a careful elicitation of rewards would arguably lead in many cases to a reward matrix more general than $\mathbb{I}$. Such a widespread use has probably been favored by the simple interpretation of predictive accuracy; a more substantial reason could be that the predictive accuracy is particularly convenient to make extensive comparisons of classifiers over many data sets, which is a key component of the machine learning practice. Accordingly, in this paper we focus on the 0-1 valued reward matrix $\mathbb{I}$.

So far we have introduced the traditional view of classification, where the predictions issued by (precise) classifiers are made of single classes. This view has been generalized through the introduction of credal classifiers [20, 22]. A *credal classifier* is also a function learned from set $\mathscr{L}$, but it maps the attributes of an instance into a set $\mathscr{K} \subseteq \mathscr{C}$ of $k := |\mathscr{K}|$ classes in general. We call this a set-valued classification. We also say that the classification is *determinate* when $k = 1$, and *indeterminate* otherwise. When a classification is fully indeterminate, that is, when $\mathscr{K} = \mathscr{C}$, we call it *vacuous*. Similarly, the *vacuous classifier* is the one that always issues vacuous predictions. To each credal classifier it is possible to associate a determinate classifier that outputs predictions by choosing every time a class uniformly at random[1] from the output set $\mathscr{K}$ of the credal classifier. We call this the *$\mathscr{K}$-random classifier*; when the related credal classifier is the vacuous one, we just call it the *random classifier*.

Evaluating a credal classifier can be regarded as the problem of defining an 'extended' reward matrix, which associates a reward gamble to each non-empty subset of classes. For instance, suppose that one believes that the vacuous and the random classifiers should be evaluated equally in a binary classification problem characterized by matrix $\mathbb{I}$. The corresponding extended reward matrix is given in Table 2.

		Actual class	
		1	2
Predicted class	1	1	0
	2	0	1
	$\{1,2\}$	0.5	0.5

Table 2: An extended reward matrix for a binary classification problem featuring reward matrix $\mathbb{I}$. This specific extended reward matrix, which values the random and the vacuous classifiers the same, originates the metric called discounted accuracy in [5] (see also Section 4). Observe that in problems with more than two classes, there will obviously be more rows to fill with the appropriate rewards gambles.

3 Introducing the betting framework

In order to make the comparison of credal classifiers as objective as possible, we introduce the idea of a betting framework. We define the framework for a traditional problem of classification, where classifiers issue determinate predictions. In Section 4 we will extend the framework to credal classification.

In the framework under consideration, we have two classifiers, which we would like to compare, that have already been inferred from data (so that there is no further learning, only an evaluation stage). These

[1] Throughout the paper we use the word 'random' to mean *uniformly* random.

classifiers are regarded as bettors. Bets correspond to instances of the problem of classification: a bet is set up by sampling an instance of the problem. Classifiers are required to bet by predicting the actual class of the instance, and are rewarded according to matrix $\mathbb{I}$. The process is repeated for ever, and the performance of classifiers is taken to be their predictive accuracy.

Let us make the betting framework more precise by describing the two types of actors that play a role there:

Bettors each of the two classifier we aim at comparing is regarded as a bettor.

House rewards are delivered to bettors by an artificial entity that we call House. House only accepts determinate bets, which are rewarded according to matrix $\mathbb{I}$.

These actors are characterized by clarifying their relationship with the rewards, that is, with the utility scale involved. To start with, based on the discussion made in Section 2, we can readily state our first assumption concerning the betting framework:

(A1) Utility of bettors is linear in the rewards.

This assumption simply states explicitly what is current practice in classification.

The second assumption concerns House. We want to model House as an agent whose only aim is to reward correct predictions. In other words, House should not introduce any subjective bias in the process of rewarding bettors because of a risk-averse or risk-seeking attitude; it should just be risk neutral:

(A2) Utility of House is linear in the rewards.

4 Betting with credal classifiers

Now we would like to extend the betting framework to credal classifiers. The crucial point here is that House only accepts determinate bets, while a credal classifier outputs set-valued classifications in general. Therefore, if we want to allow a credal classifier to play, we should find a way to extend the reward matrix to set-valued classifications in a way that both House and bettor find acceptable.

The first step in this direction is to recognize that any negotiation between the credal classifier and House can be made only on the basis of determinate bets, which is the only language that House understands. In order to enable the credal classifier to play as a determinate bettor, we state the following assumption:

(A3) The credal bettor accepts betting on any single class from its set-valued prediction, if forced to make a determinate bet, and on no class outside that set.

This assumption is typically satisfied when the classes in the output set of the credal classifier are incomparable, and the other ones represent dominated options. This is the case when credal classifiers are obtained using sets of probabilities and decision criteria like maximality or e-admissibility (see, e.g., [19, Section 3.9]). We state the assumption explicitly in order to allow the framework to be used also by credal classifiers created in a different way.

The next assumption formalizes the idea that the framework is run for ever:

(A4) Every possible bet is repeated infinitely many times in the betting framework by sampling the problem instances.

This assumption, together with the previous one, enable the credal classifier to actually adopt a randomized strategy over the k classes in its output set $\mathscr{K}$. A randomized strategy is a mass function $\sigma := (\sigma_i)_{i \in \mathscr{K}}$ that represents the (determinate) betting behavior of the credal classifier in the limit.

At this point House knows that the credal classifier has the freedom to implement any randomized betting strategy: this means that the credal classifier can actually force House to undergo any expected loss that can follow from the choice of the strategy.

Let us call a prediction $\mathscr{K}$ 'successful' if the actual class belongs to $\mathscr{K}$. We restrict the attention to successful predictions as they determine House's expected loss: in fact, an unsuccessful prediction always yields a zero loss, by definition of $\mathbb{I}$, irrespective of the randomized strategy adopted. Let $\theta := (\theta_j)_{j\in\mathscr{C}}$ be the vector of chances, that is, the population proportions, for the classes conditional on the prediction being successful (this means that $\theta_j = 0$ if $j \notin \mathscr{K}$). House's expected loss conditional on a successful predictions equals

$$\sum_{i\in\mathscr{K}} \sum_{j\in\mathscr{C}} \mathbb{I}_i(j)\sigma_i\theta_j = \sum_{i\in\mathscr{K}} \sigma_i\theta_i,$$

where we are assuming that the strategy is chosen independently of the chances.

The loss depends on σ, which is chosen by bettor, and on θ. The latter models the specific problem under consideration. But House knows that the betting system will be applied, in principle, to every possible problem. House should then be enabled to consider every possible scenario:

(A5) In the determination of the expected loss, House has the freedom to choose any value for θ.

At this point we are ready to derive the extended reward matrix (as described at the end of Section 2, and in particular given in Table 2 for the case of binary classification):

Theorem 1. *Let $\mathscr{K} \subseteq \mathscr{C}$ be a set-valued prediction made of k classes, $\mathbb{I}_\mathscr{K}$ be the indicator function of set $\mathscr{K}$, and j the actual class. The corresponding value in the extended reward matrix that is uniquely consistent with (A1)–(A5) is the discounted accuracy:*

$$\frac{\mathbb{I}_\mathscr{K}(j)}{k}. \tag{1}$$

Proof. If $\mathscr{K}$ is unsuccessful, then any randomized strategy will yield a zero loss. Let us focus on successful predictions. Let Δ be the $n-1$ probability simplex. We formulate the problem in a game-theoretic setting. The two players are just bettor and House. Bettor can choose $\sigma \in \Delta$, while House can choose $\theta \in \Delta$. What we get is a zero-sum game with a gain for bettor defined by $\sum_{i\in\mathscr{K}} \sigma_i\theta_i$. This is a continuous linear function in σ for all $\theta \in \Delta$, as well as in θ for all $\sigma \in \Delta$, and moreover Δ is a compact convex set. The minimax theorem (see, e.g., [17, Theorem 6.7.3]) allows us to deduce that there is an optimal solution to the game with expected reward equal to $\max_{\sigma\in\Delta}\min_{\theta\in\Delta}\sum_{i\in\mathscr{K}} \sigma_i\theta_i$. It is easy to see that that is equal to $1/k$: once a strategy σ is fixed, the minimum is achieved by setting $\theta_{i_*} := 1$ on any $i_* = \operatorname{argmin}_{i\in\mathscr{K}} \sigma_i$; then the problem becomes $\max_{\sigma\in\Delta}\min_{i\in\mathscr{K}} \sigma_i = 1/k$. The related optimal strategy σ^* is uniform, $\sigma^*_i := 1/k$ for all $i \in \mathscr{K}$; this means that bettor and House agree that credal bettor should act like the $\mathscr{K}$-random classifier.

Now remember that, according to (A1)–(A2), both bettor and House are risk neutral. This means they agree that an unsuccessful prediction is rewarded by the certain value 0 and a successful one by the certain value $1/k$. This is achieved by setting the reward equal to the discounted accuracy. □

It is useful to comment on this result from a few different viewpoints.

One thing is that the discounted accuracy implements a kind of least-committal reward system for House, in the sense that House gives bettor only what is certainly due to it. In fact, if the credal bettor does implement strategy σ^*, the expected reward that it achieves is indeed $1/k$, irrespective of the chances. Therefore the established reward is what House knows already that bettor can make for sure. For the same reason, it would be implausible to expect that credal bettor accepts any smaller reward. It is also interesting to observe that playing as the $\mathscr{K}$-random bettor (i.e., classifier) is the only way for credal bettor to have a sure reward.

The next consideration is again based on the observation that credal bettor is evaluated exactly as the $\mathscr{K}$-random bettor. This has important implications for the comparison of classifiers through the discounted accuracy: the main point is that the $\mathscr{K}$-random bettor is actually taken as a baseline to compare classifiers. Consider, for the sake of explanation, a determinate classifier whose output class is always contained in that of a certain credal classifier. The determinate classifier will be evaluated better than the credal classifier as

soon as it exploits, to any (even a very tiny) degree, the credal classifier's set of output classes better than the $\mathscr{K}$-random one. Looking at this from another side, it means that the credal classifier can be better than the determinate one only if the latter behaves worse than the $\mathscr{K}$-random classifier! (Surprisingly, this does not happen as rarely as one could imagine; see Section 9.1.3 for details.) This discussion should make clear that the discounted accuracy, although it is a reasonable criterion, is probably the most unfavorable way (among the reasonable ones) to evaluate credal classifiers, as a credal classifier cannot do better than isolating a set of classes that is impossible to compare.

This points to an aspect of the evaluation that the discounted accuracy certainly fails to capture. Let us focus on the simplest possible setup, using the following example. You are trying to evaluate two physicians based on some recorded diagnostic performance of theirs. In your records, the first physician always issues a vacuous diagnosis, that is, the entire set $\mathscr{C}$ of possible diseases. The second always issues a determinate diagnosis. But when you measure the second physician's predictive accuracy, you realize that his predictions are random. In this case, the discounted accuracy values the two physicians the same: $1/n$. But it is clear that the first physician provides you with something more than the second, because, in a sense, he delivers what he promises. How to precisely value this 'something more' appears to be quite a subjective matter. In this sense, it should not be too surprising that discounted accuracy does not value it at all, as it has been created trying to keep subjectivity out of consideration. And yet, subjectivity matters, and should be taken into account. The next section shows that this can be done in a very natural way.

5 Comparing credal classifiers

We have two classifiers f, g. We focus on selecting the classifier whose expected performance in the next instance (i.e., next bet) is greater than the other's. In the previous section we have measured performance by discounted accuracy. In this section, we want to make the method of comparison more flexible by allowing subjectivity to enter the picture, so as to be able to deal with the issues discussed at the end of the previous section. To this end, we start identifying classifiers with gambles: let gambles f and g yield the discounted-accuracy reward achieved by classifiers f and g, respectively, in the next instance. There is uncertainty about these gambles because we assume that the instance has yet to be sampled.

The comparison of gambles f and g needs a (rational) decision maker, whom we call 'you'. By definition of the gambles, you will compare them based on discounted-accuracy rewards. We model your attitude towards these rewards through the following assumption:

(A6) Your utility,[2] as a function $u(\cdot)$ of the discounted-accuracy rewards, is concave,

which means that you are risk averse, or at most neutral, in these rewards.[3]

This seems to be quite a reasonable assumption in the common setup where the original rewards (the ones used to define the 0-1 reward matrix $\mathbb{I}$) are measured in a utility scale that is linear for you. To see this, imagine that you are directly asked to extend the original reward matrix $\mathbb{I}$ to take into account your attitude towards set-valued classifications. Can we say something about the values you would use to define such an extended matrix? On the one hand, we argue that the rewards you would put there should be greater than or equal to the discounted-accuracy rewards. This follows from the discussion at the end of Section 4, which shows that it would be unreasonable to use values smaller than the discounted accuracy. On the other hand, values strictly greater than that would be reasonable: these allow you to express a preference in favor of a set-valued classification in comparison to the related $\mathscr{K}$-random prediction.

[2] We assume that the usual regularity conditions for utility hold, and in particular that it is strictly increasing, and that it has first and second derivatives (see, e.g., [15]).

[3] Note that House is not affected by your entering the picture, as it keeps on delivering discounted accuracy rewards as before. What changes is the explicit introduction of a decision maker and his perception of the value of these rewards, as modeled by your risk aversion.

Remember that your utility is linear in the original 0-1 rewards, whence we can assume without loss of generality that for you it holds that $u(0) = 0$ and $u(1) = 1$. In addition, the previous argumentation makes it clear that for a set-valued prediction $\mathscr{K}$ containing the actual class, it should only be possible that

$$u(1/k) \geq 1/k.$$

In other words, your utility function turns out to be, in general, a non-linear function of the discounted accuracy rewards. Saying this differently, we can equivalently regard discounted accuracy as representing a new utility scale in which your utility function is non-linear. In Assumption (A6) we take your utility in particular to be concave to express a consistent preference for set-valued classifications in comparison to the related $\mathscr{K}$-random predictions (note that this includes the extreme case of a linear utility function, in which the two options are equally valued).

Going back to the comparison of classifiers, it follows immediately from (A6) and decision-theoretic arguments that you will choose the one with maximum expected utility: $h^* := \operatorname{argmax}_{h \in \{f,g\}} E[u(h)]$.

Re-consider the example of the vacuous and the random classifier, discussed at the end of Section 4, as they are emblematic of the differences that arise in the evaluation of credal and precise classifiers when using utility.

Proposition 1. *The random and the vacuous classifiers have the same expected reward on the next instance, but the expected utility of the vacuous is greater under any strictly concave utility function.*

Proof. Denote the random classifier by r, and the vacuous classifier by v. As usual, we identify the classifiers with the corresponding gambles, which represent uncertain discounted-accuracy rewards for the next bet. The vacuous classifier gets on any instance the deterministic reward $1/n$. Thus, under any utility function:

$$E[u(v)] = u\left(\frac{1}{n}\right) = u(E[v])\,.$$

The random classifier r samples the predicted class from $\mathscr{C}$ according to the uniform mass function σ^*, independently of the actual class. Let us denote, as usual, by $\theta = (\theta_j)_{j \in \mathscr{C}}$ the vector of chances for the actual classes. We obtain that

$$E[r] = \sum_{i \in \mathscr{C}} \sum_{j \in \mathscr{C}} \mathbb{I}_i(j) \sigma_i^* \theta_j = \sum_{i \in \mathscr{C}} \sigma_i^* \theta_i = 1/n\,.$$

This shows that $E[v] = E[r]$. In addition, using Jensen's inequality leads to

$$E[u(r)] < u(E[r]) = u\,(1/n) = E[u(v)]\,,$$

whenever u is a strictly concave function. □

To better analyze this point, it is useful to approximate the expected utility by a second-order Taylor series. Let h be a generic classifier (and hence, a gamble):

$$\begin{aligned} E[u(h)] &\simeq u(E[h]) + \overbrace{u'(E[h])E(h - E[h])}^{=0} + \\ &+ \frac{1}{2} u''(E[h]) E\left[(E[h] - h)^2\right] = \\ &= u(E[h]) + \frac{1}{2} u''(E[h]) Var[h]\,, \end{aligned} \tag{2}$$

where u', u'' are the first and second derivative of the utility function, and $Var[h]$ denotes the variance of h. Well-known papers in finance [12, 14] have shown that this is a very accurate approximation.

Remember that $u''(E[h]) \leq 0$ for every concave utility function (moreover, $u''(\cdot)$ is related to the degree of risk aversion of the utility assessor). Therefore what Equation (2) tells us is that the expected utility increases by increasing the expectation of rewards and decreasing their variance. It is clear now why the vacuous classifier, with variance equal to zero, is preferred to the random one. In other words, the 'something more' that the vacuous classifier is providing is its inherent reliability in earning rewards, which, using discounted accuracy, has a very clear numerical counterpart in its variance. The value that you give to this is indeed personal, and is formalized through your utility function. In the extreme case when you are risk neutral in the discounted-accuracy rewards, the value is zero, and in this case there seems to be little room for credal classifiers in your interests. Bigger values express stronger preferences for reliable predictions.

All the above considerations can be turned into a remarkably simple procedure to empirically compare credal classifiers in practice. Remember that in a classification problem we usually have a test set $\mathscr{T}$, that is, a collection of instances used to evaluate the performance of a classifier. We need to estimate $E[u(h)]$ for a certain classifier h. Let us denote by $\mathscr{U}$ the set of values that gamble $u(h)$ can take. Set $\mathscr{U}$ has $(2^n-1)\times n$ elements at most, as the values are in one-to-one correspondence with the elements of the reward matrix extended through discounted accuracy (as in Table 2). If we estimate the chance of a value $u_h \in \mathscr{U}$ by its sample proportion $\#(u_h)/|\mathscr{T}|$ in the test set, we obtain:

$$E[u(h)] \simeq \sum_{u_h \in \mathscr{U}} u_h \frac{\#(u_h)}{|\mathscr{T}|} = \frac{1}{|\mathscr{T}|} \sum_{(a,c)\in\mathscr{T}} u(h(a,c)).$$

This is equivalent to evaluating the performance of a credal classifier using the $(2^n-1)\times n$ reward matrix obtained by applying function $u(\cdot)$ point-wise to the matrix extended through discounted accuracy. As an example, the extended reward matrix in Table 2 should be transformed into that in Table 3. In other

		Actual class	
		1	2
Predicted class	1	u(1)	u(0)
	2	u(0)	u(1)
	$\{1,2\}$	u(0.5)	u(0.5)

Table 3: The extended reward matrix in Table 2 modified through your utility of discounted accuracy.

words, what is done in practice is to change the 'discounting' factor in the discounted accuracy by means of the concave utility function. For this reason, we call the resulting metric 'utility-discounted (predictive) accuracy'.

As a side remark, consider that the performance index obtained through the application of utility to the extended reward matrix need not be in the range $[0,1]$ in general; in this case it cannot be interpreted as an accuracy index. To bring the measure back to a predictive accuracy index (although one that is biased through the utility function in order to take into account your personal preferences), the above comparison can be made, more conveniently, using $u^{-1}(E[u(h)])$ (that is, by the so-called *certainty equivalent*).

6 A digression: betting with money

Before moving on to more technical questions, it is useful to look back one more time at the utility-discounted accuracy we are proposing. We do this by briefly overviewing an alternative interpretation of the metric that offers a different viewpoint.

We started the discussion in Section 2 assuming that the rewards are measured in a utility scale that is linear for you. We did so because this appears to be the standard assumption done in classification (even though it is often left implicit). On the other hand, the development of utility-discounted accuracy can be made simpler by assuming that the rewards are given using money. This means assuming that a successful

determinate prediction earns you 1 unit of a given currency (such as dollars), and that you earn nothing otherwise.

Using money simplifies the treatment because it allows us to focus only on House. House is, as before, an artificial entity that delivers rewards—this time using money—. House's aim is only to reward correct rewards, and for this reason Assumption (A2) still holds. Initially House sets the rewards for the determinate bets, thus making up matrix $\mathbb{I}$. Then House wants to allow for set-valued bets, and wonders how to associate a reward to them in a fair way. The discussion in Section 4 can be re-phrased to this end so as to show that the discounted accuracy implements the fair way to reward bettors that House is looking for: in fact, from House's viewpoint, it would be unfair to reward a bettor with less than the discounted accuracy, because the bettor could obtain the same average amount of money in the limit by playing as the $\mathscr{K}$-random bettor; on the other hand, delivering more than the discounted accuracy would give an unfair advantage to the bettor, because there is no way for him to surely make a gain bigger than the discounted accuracy in general.

By doing so, House fixes the rewards for any type of bet, determinate and indeterminate. At this point, you want to bet with House. Remember that House gives rewards in money. It is a widely accepted assumption that people are risk averse in money rewards. This means that your utility, as a function of money, is concave; whence Assumption (A6), properly re-phrased for the present setup, holds. The rest of the discussion in Section 5 holds here as well.

In other words, in the money-based framework, utility-discounted accuracy follows from two quite natural requirements: setting up a fair way to reward set-valued predictions on the basis of the rewards of determinate ones; and taking into consideration that you are risk averse in money rewards. Apart from the inherent simplicity of the money-based setup, it is remarkable that such a setup makes it immediate clear that there is a large overlap between the evaluation of credal classifiers and finance. In this light, it is not surprsing that a well-known formula in finance, such as (2), holds also in the present setup. In finance, such a formula states that an investor usually wants to decrease the risk of an uncertain gain besides increasing the gain itself. With credal classifiers, such a formula states that one might want to go for weaker predictions in order to increase their reliability. Underlying both views is the idea that there is a value in reducing variability: using indeterminate predictions seems to be a privileged way to do so in classification.

We end up the discussion about money rewards here; in the next sections we take up again the main thread of the discussion, concerned in particular with rewards expressed in a utility scale that is linear for you. This notwithstanding, it is useful to keep in mind that also the next developments can be re-casted in terms of money-based rewards.

7 The case for an objective winner

Equation (2) is useful because it gives us a very accurate approximation to the expected utility while releasing us from having our considerations narrowed down by the specific form of the utility function considered. To this end, in the following, we will repeatedly refer to (2) as if it were our actual expected utility.

In particular, an interesting consideration suggested by Equation (2) is that in one case the comparison of classifiers can be done by minimizing subjective considerations: when the two classifiers have equal expected reward. In this case, the classifier with minimum variance wins under every strictly concave utility function: that is, no matter how tiny (but non-zero) is your degree of risk aversion. This can be implemented in practice by defining a range where the difference of the expected rewards is deemed irrelevant, and estimating their variances from the test set.

In the following, we investigate whether we can relate the variance of a classifier with its *determinacy*, that is, with a measure of the amount of imprecision in the output. Intuitively, we expect such a relationship to exist because both measures are related to the reliability of a classifier, and moreover, we expect that larger indeterminacy corresponds to smaller variance.

The gamble h corresponding to a classifier's performance in the next bet can be decomposed in two other gambles h_D and h_I such that $h = h_D + h_I$ and $h_D h_I = 0$ (element-wise). Intuitively, h_D and h_I represent the

rewards for h when it returns, respectively, a determinate and an indeterminate classification. The following relationships follow from the decomposition under discounted accuracy:

$$E[h^2] = E[h_D^2] + E[h_I^2], E[h_D^2] = E[h_D], E[h_I] \geq E[h_I^2],$$

where in the last expression we have the equality only if $E[h_I] = E[h_I^2] = 0$, which implies that either h is a precise classifier or that indeterminate predictions of h contain the actual class with probability zero.

Let f and g denote two generic classifiers with the same expected discounted accuracy: $E[f] = E[g]$. Using the identities above, one can show that the difference of variances is thus

$$\Delta Var \coloneqq Var[g] - Var[f] = E[g_D] + E[g_I^2] - E[f_D] - E[f_I^2]. \tag{3}$$

Let us start by considering the important case where we compare a credal classifier with a precise one:

Proposition 2. *Consider a credal classifier and a precise classifier with the same expected reward. Then the credal classifier is preferable to the precise classifier under any strictly concave utility function.*

Proof. Let us denote by f the credal classifier and by g the precise one. We know by Equation (2) that we prefer the classifier with smaller variance under any strictly concave utility function. Thus, it suffices to show that $\Delta Var \geq 0$. Since $E[f_I^2] \leq E[f_I]$, it follows from Equation (3) that $\Delta Var = E[g_D] - E[f_D] - E[f_I^2]$ so that

$$\Delta Var \geq E[g_D] - E[f_D] - E[f_I] = E[g] - E[f],$$

which equals zero, since f and g have equal expected reward. Note the inequality is strict (i.e., there is strict preference) if the credal classifier is not always determinate and its indeterminate predictions are successful with positive probability. □

Now, let H_D be the event that equals 1 when the generic classifier h is determinate on the next instance, and 0 otherwise. We define the *determinacy* of classifier h as the probability that h is determinate: $P(H_D)$. This definition allows us to settle the problem for the next case:

Proposition 3. *Consider two credal classifiers that are vacuous whenever they are indeterminate and that have the same expected reward. Then the more indeterminate classifier is preferable under any strictly concave utility function.*

Proof. Let us denote by f and g the two credal classifiers, assuming f to be more indeterminate than g: $P(G_D) > P(F_D)$. It suffices to show that $\Delta Var > 0$. Any generic classifier h that is vacuous whenever it is indeterminate is rewarded with $1/n$ for any indeterminate prediction. Hence,

$$E[h_I] = \frac{1 - P(H_D)}{n}, E[h_I^2] = \frac{E[h_I]}{n}.$$

From these identities and Equation (3) we have that

$$\begin{aligned}
\Delta Var &= E[g_D] + E[g_I]/n - E[f_D] - E[f_I]/n \\
&= -E[g_I] + E[g_I]/n + E[f_I] - E[f_I]/n \\
&= \frac{n-1}{n}\left(-E[g_I] + E[f_I]\right) = \frac{n-1}{n^2}\left(P(G_D) - P(F_D)\right),
\end{aligned}$$

which is strictly positive by the initial assumptions. □

This proposition is particularly useful as it allows us to solve the problem in the case of binary classification problems, where any indeterminate prediction is necessarily vacuous.

One might be tempted to think that the previous result extends to non-vacuous classifiers as well, that is, the more determinate a classifier the higher its variance (and therefore the less preferable it is). Unfortunately, this is not the case, as the following example shows.

Example 1. Consider a three-class classification problem. Let H_k denote the event that equals 1 if the generic classifier h returns a set of k classes that contains the actual one, and 0 otherwise. Likewise, let H_k^c be the event that equals 1 if h outputs k incorrect classes, and 0 otherwise. Note that $\sum_{k=1}^{3} H_k + H_k^c = 1$ and $H_3^c = 0$. We can define the relevant expectations in terms of H_k, H_k^c:

$$P(D_h) = P(H_1) + P(H_1^c), \qquad E[h] = \sum_{k=1}^{3} \frac{1}{k} P(H_k),$$

$$E[h^2] = \sum_{k=1}^{3} \frac{1}{k^2} P(H_k), \qquad 1 = \sum_{k=1}^{3} P(H_k) + P(H_k^c).$$

Assume that $P(F_1) = P(G_1) + \varepsilon$, $P(G_1^c) = P(F_1^c) + 2\varepsilon$, $P(G_2) = P(F_2) + 2\varepsilon$, $P(F_2^c) = P(G_2^c) + 3\varepsilon$, and $P(F_3) = P(G_3)$, for some small $\varepsilon > 0$. Then we have from the identities above that $E[f] = E[g]$. Similarly, we have that $E[f^2] = E[g^2] + \frac{\varepsilon}{2}$. Hence, $\Delta Var = E[g^2] - E[f^2] < 0$, and g is preferred over f even though g is more determinate than f: $P(D_f) = P(D_g) - \varepsilon$.

Alternatively, we might measure the indeterminacy of a classifier h by the expected number of classes it outputs: $\sum_{k=1}^{n} k \left[P(H_k) + P(H_k^c)\right]$. Thus, in the example, we would have

$$\sum_{k=1}^{n} k\,[P(F_k) + P(F_k^c)] = \sum_{k=1}^{n} k\,[P(G_k) + P(G_k^c)] + 4\varepsilon,$$

and g is preferred over f even though the former has a smaller expected number of outputs than the latter. ♦

8 Comparison over the next m bets

So far, we have considered the expected reward and utilities for the next *single* classification; this setting fits for instance the case of a patient, who asks a doctor for a diagnosis and who is concerned only about the utility generated by the very next classification (his diagnosis). Conversely, an online trader, who performs m trading operations every day, might accept to lose some money in the very next transaction, provided that the set of m transactions generated at the end of day has high enough utility. In this case, expected rewards and expected utilities should be computed over the next m bets. In the following, we compare the random classifier r and the vacuous classifier v on the next m bets; we denote by v_m and r_m the rewards of the vacuous and the random ones over the next m instances.

Gamble v_m has deterministic value m/n and thus:

$$E[u(v_m)] = u\left(\frac{m}{n}\right).$$

To compute $E[u(r_m)]$, let us consider that classifier r yields utility $u(\ell)$ when it correctly predicts ℓ outcomes in the next m bets; considering that classifier r issues a correct classification with probability $1/n$ (see Proposition 1), the probability of correctly predicting ℓ instances out of the next m is the binomial:

$$Bin(\ell, m, \frac{1}{n}) = \binom{\ell}{m} \frac{1}{n}^{\ell} \left(1 - \frac{1}{n}\right)^{m-\ell}.$$

The expected utility produced by the random classifier over the next m bets is thus:

$$E[u(r_m)] = \sum_{\ell=1}^{m} u(\ell) Bin(\ell, m, \frac{1}{n}). \tag{6}$$

It is not immediate to compare the expected utilities of the random and vacuous classifiers using Equation (6); a clear understanding can be obtained through the second-order approximation given by Equation (2). In the following, we analyze in this way the logarithmic and the exponential utility. The second-order approximation of both the logarithmic and the exponential utility is very good, having relative absolute error consistently smaller than 1%.

8.1 Logarithmic Utility

The logarithmic utility is $u(x) := \log(1+x)$, whence $u''(x) = -\frac{1}{(1+x)^2}$; applying Equation (2), we get:

$$u(E[r_m]) + \frac{1}{2}u''(E[r_m])Var(r_m) = u(E[r_m]) - \frac{Var(r_m)}{2(E[r_m]+1)^2} = u\left(\frac{m}{n}\right) - \frac{m\frac{1}{n}\left(1-\frac{1}{n}\right)}{2(\frac{m}{n}+1)^2},$$

where in the last passage we introduced the analytical expression of the variance for a binomial distribution.

Thus, the (approximated) difference between the expected utility of the random and the vacuous over the next m bets is

$$d(m) = E[u(v_m)] - E[u(r_m)] = \frac{\frac{m}{n}\left(1-\frac{1}{n}\right)}{2(\frac{m}{n}+1)^2} \propto \frac{m}{(\frac{m}{n}+1)^2}, \tag{7}$$

where in the last passage we removed the proportionality constant $\frac{1}{2n}\left(1-\frac{1}{n}\right) > 0$. Function $d(m)$ is shown in Figure 1.

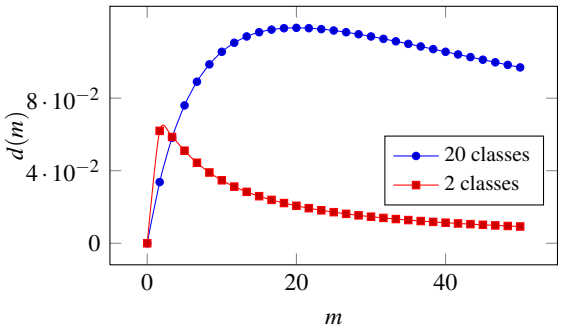


Figure 1: Function $d(m)$ for *logarithmic* utility, under different number of classes.

The first derivative of $d(m)$ is:

$$d'(m) = \frac{1}{(\frac{m}{n}+1)^2} - 2\frac{\frac{m}{n}}{(\frac{m}{n}+1)^3} \propto 1 - \frac{m}{n}, \tag{8}$$

where the last passage is obtained considering that $(\frac{m}{n}+1)^3 > 0$. From Equations (7) and (8), we can figure out that $d(m)$ will monotonically increase up to $m < n$ (*inversion point*), to then indefinitely decrease, so that $d(m) \to 0$ for $m \to \infty$; if expectations of utilities are computed over a long enough number of bets, the expected utility produced by the two classifiers is the same. It also follows that increasing n delays the convergence of the expected utilities to the same value, as also shown in Figure 1.

8.2 Exponential Utility

The exponential utility is $u(x) \coloneqq 1 - \exp(-ax)$, where a is a coefficient of risk aversion. Noting that $u''(x) = -a^2 \exp(-ax)$, the second-order approximation yields:

$$u(E[r_m]) + \frac{1}{2}u''\left(\frac{m}{n}\right) Var(r_m) =$$
$$u\left(\frac{m}{n}\right) - \frac{1}{2}a^2 \exp\left(-a\frac{m}{n}\right) m\frac{1}{n}\left(1-\frac{1}{n}\right),$$

whence

$$d(m) = -\frac{1}{2}a^2 \exp\left(-a\frac{m}{n}\right) m\frac{1}{n}\left(1-\frac{1}{n}\right) \propto$$
$$\propto -\exp\left(-a\frac{m}{n}\right) m,$$

where the proportionality constant is $\frac{a^2}{2}\frac{1}{n}\left(1-\frac{1}{n}\right) > 0$.

We have

$$d'(m) = \exp\left(-a\frac{m}{n}\right) \cdot \left(a\frac{m}{n} - 1\right).$$

Function $d(m)$ has qualitatively the same behavior of the logarithmic case, but the inversion point is now located at $m = \frac{n}{a}$. Moreover, the difference between the expected utility of the two classifiers depends also on the risk-aversion coefficient a; higher risk aversion delays the convergence of the expected utilities, thus emphasizing the difference in favor of the vacuous on small m.

9 Experiments

To experimentally assess the performance of a credal classifier, we need a utility function to be applied on top of discounted accuracy. Let us fix the utility of a correct and determinate classification as $u(1) \coloneqq 1$ and the utility of a wrong classification (determinate or indeterminate) as $u(0) \coloneqq 0$; this is consistent with the assumption that you are risk neutral in the scale of the original rewards.

Let us initially consider the case of a binary classification problem. To fully specify the utility function, we have to define the value of $u(0.5)$. With $u(0.5) = 0.5$, the utility leads to the standard discounted accuracy measure. Risk-aversion increases as $u(0.5)$ increases. We assume that in case of risk aversion, $u(0.5)$ can reasonably lie between 0.65 and 0.8; we call u_{65} and u_{80} the utility functions corresponding to these choices.

In order to deal with classification problems with more than two classes, we adopt a quadratic utility function, which passes through $u(0) = 0, u(1) = 1$, and $u(0.5) = 0.65$ or $u(0.5) = 0.80$. The two utility functions are:

$$u_{65}(x) = -1.2x^2 + 2.2x,$$
$$u_{80}(x) = -0.6x^2 + 1.6x,$$

where x denotes the discounted accuracy of the issued classification. These functions are shown in Figure 2. Note that u_{80} is slightly greater than 1 for some x between 0.5 and 1; however, this part of the function is never used, as discounted accuracy cannot assume values between 0.5 and 1. This means that the range of values returned by all the functions will be $[0,1]$, which will allow us to interpret the resulting utility-discounted accuracies indeed as predictive accuracies (even though they are biased accuracies so as to take into account your personal preferences).

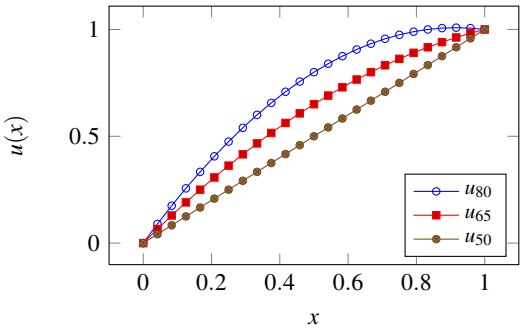


Figure 2: The three quadratic utility functions obtained for different values of $u(0.5)$. The function corresponding to discounted accuracy is u_{50}.

A well-known drawback of the quadratic utility function is that it models the risk aversion as increasing with wealth. Yet, at least under particular conditions, quadratic and exponential utility result in very similar choices [1]. Moreover, using quadratic utility has the advantage to make Eq. (2) an exact representation of the utility function, whose effects can then be interpreted very clearly as due to a compromise between increasing gain and reducing variability. We considered using the exponential utility too, but we eventually discarded it as it could not satisfactorily fit all the three values we chose for $u(0.5)$.

9.1 NBC versus NCC

In this section we focus the comparison on two well-known classifiers: naive Bayes classifier (NBC [8]) as a representative of precise classifiers, and the naive credal classifier (NCC [20, 21, 4]). In particular, we compare the expected utility generated by NBC and NCC on the next *single* bet, namely on a single instance.

9.1.1 Artificial data sets

In a first set of experiments, we generated artificial data sets considering a binary class and 10 binary features; we set the marginal chances of classes as uniform, while we drew the conditional chances of the features under the constraint $|\theta_{i1\ell} - \theta_{i2\ell}| \geq 0.1 \ \forall i, j$, where $\theta_{ij\ell}$ denotes the chance of feature A_i to be in state ℓ when $C = j$; the constraint forced each feature to be strongly dependent on the class. We drew θ 80 times uniformly at random and we consider the sample sizes: $s \in \{25, 50, 100\}$. We did not consider larger sample sizes, under which NCC would have been almost completely determinate, and thus not really different from NBC. For each pair (θ, s) we generated 50 training sets; we then evaluate the trained classifiers on a test set of 10000 instances. In the following, the instances indeterminately classified by NCC are referred to as the *area of ignorance*. For each sample size, we thus perform $80\theta \times 50$ trials $= 4000$ training/test experiments. In these experiments, we only consider the u_{65} utility function.

As can be seen in Figure 3, the discounted accuracy of NBC is higher than the discounted accuracy of NCC; this means that, on the area of ignorance, NBC is doing better than the $\mathscr{K}$-random guesser, namely the classifier which picks the class at random from among those returned by NCC. Note that for NBC there is no distinction between accuracy and utility. However, NCC produces higher expected utility than NBC at each sample size, even under the conservative choice u_{65}.

9.1.2 Experiments with downsampling

We then performed some experiments on the kr-kp data set from the UCI repository. It is a binary data set, in which the two classes are evenly distributed; it contains 36 binary features and 3200 instances. We

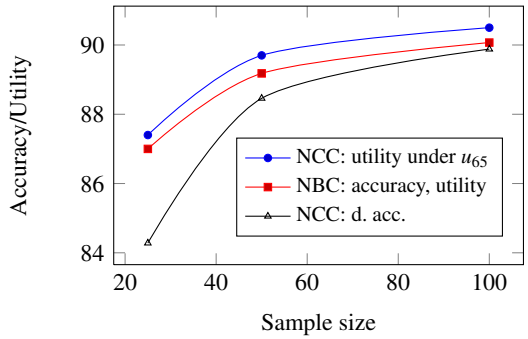


Figure 3: Experimental results with artificial data; each point shows the median over 4000 experiments, performed with the same sample size s. For NBC, accuracy and utility coincide.

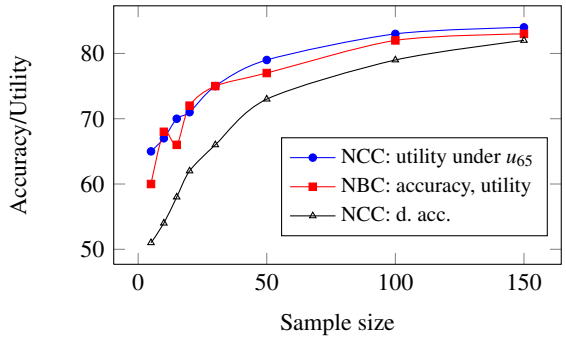


Figure 4: Utility and discounted accuracy generated by NBC and NCC for downsampled versions of kr-kp.

downsampled the data set, generating training sets of size $s \in \{5, 10, 15, 20, 30, 50, 100, 150\}$. For each sample size, we sampled 100 different training sets; the test set was given by the instances left in the original data set. Again, we considered the conservative choice u_{65}. The average results are shown in Figure 4. The determinacy of NCC (not shown) steadily increases with the sample size, as well as the discounted accuracy of NCC and the accuracy of NBC. For very small sample sizes, NCC is almost always indeterminate; in this case, its utility corresponds to $u(0.5)$ and thus is 0.65; in the same situation NBC is only slightly better than random guessing. Both the expected utility of NBC and NCC increases with the sample size; that of NCC remains however slightly superior. By adopting the u_{80} function instead of the u_{65}, the advantage of NBC would have obviously been more evident.

9.1.3 Experiments with real data sets

To get a more comprehensive picture of the performances of NBC and NCC, we performed experiments with 55 datasets from the UCI repository. We will consider this collection of data sets also in the next sections. On each data set, we performed ten runs of ten-fold cross validation. To compare two classifiers on the *whole* collection of data sets, we used the Wilcoxon signed-rank test, as recommended by [6]. All tests are performed with $\alpha := 0.05$.

Since NCC operates only on discrete data, each dataset has been discretized using the MDL-based discretization. Under the risk-neutral utility (i.e., discounted accuracy), the accuracy of NBC was higher than the discounted accuracy of NCC in 33 of the datasets, while the converse held on 23 datasets; this difference was statistically significant. This indicates that NBC most often performs better than the $\mathscr{K}$-random guesser

on the instances where NCC is indeterminate; surprisingly, it also happens to perform worse in several data sets. However, no significant difference was found under u_{65} and under u_{80}, when comparing the expected utility produced by NCC and NBC over the collection of data sets. The expected utilities of NBC and NCC on each dataset under u_{65} and u_{80} are compared in Figure 5 through scatter plots. The straight line in the figures represents equal expected utility performance. Points above the line indicate cases where NBC outperformed NCC; conversely, points below the line indicate cases where NCC outperformed NBC. The plots show a continuous increase of the expected utility of NCC on each dataset as we move from u_{65} to u_{80}, as expected.

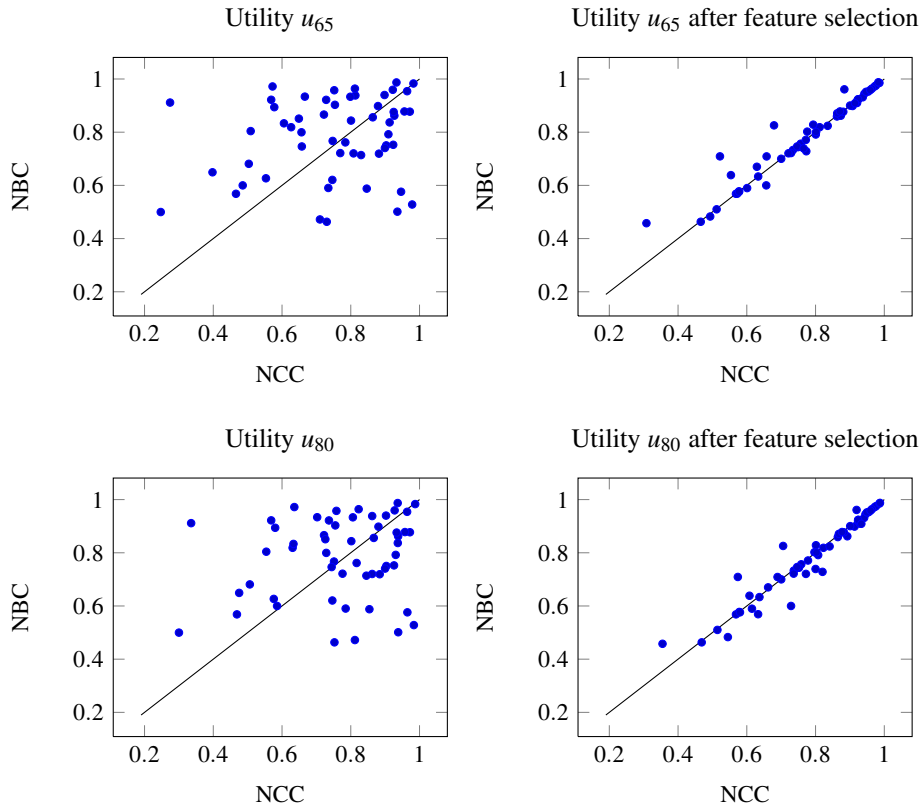


Figure 5: Scatter plot comparing NBC and NCC on the 55 datasets.

The median accuracy of NBC was 82%, while the median expected utility of NCC vary from 75% under linear utility to 78% under u_{65} to 81% under u_{80}, suggesting that, as expected, NCC's performance improves as we increase risk aversion. On 33 out of the 55 datasets, NCC had a determinacy higher than 90%, indicating that, in most of the datasets, NCC issued few indeterminate predictions. The median determinacy was 0.95.

NCC faces difficulties when the contingency table induced from a data set contains many zero counts, which causes NCC to be excessively indeterminate. Several such data sets are comprised in our collection. In these cases, restricting the credal set of NCC can largely increase the expected utility of NCC [3]; however, this is outside the scope of this paper.

However, in order to mitigate such problems, we repeated the experiments after pre-processing each datasets with a correlation-based feature selection algorithm. This is anyway a sensible approach, since naive classifiers are negatively affected by redundant features. The scatter plots after feature selection are shown

in the right column of Figure 5; after feature selection, NBC and NCC performed much more similarly. In fact, feature selection increases the determinacy of NCC: on 37 datasets NCC had a determinacy higher than 90%. Thus, it is not surprising that NBC and NCC exhibit similar performance in most of the datasets. However, NCC shows in this case a better performance than NBC; in particular:

- under u_{65}: NCC has higher expected utility in 37 data sets, NBC in 28; this difference was *not* significant;
- under u_{80}: NCC has higher expected utility in 42 data sets, NBC in 13; this difference was *significant*.

As last analysis, we compared the expected utility produced by the two classifiers only on the instances indeterminately classified. This comparison is meaningful, since on the instances determinately classified the two classifiers return the same classes. Under u_{65}, NCC produces on the indeterminately classified instances, an expected utility that is, averaging over all data sets, about 9% greater than that produced by NBC on the same instances; the 95% confidence interval of this improvement is 1–19%. Under u_{80}, NCC produces on the indeterminately classified instances, an expected utility which is, averaging over all data sets, about 36% greater than that produced by NBC on the same instances; the 95% confidence interval of this improvement is 24–48%. These values refer to the case when feature selection is applied; they show that it is beneficial to suspend the judgment on the instances identified as doubtful by NCC. Yet, feature selection seems important to exploit the maximum potential of NCC, which otherwise might suffer from excessive indeterminacy.

9.2 Experiments with credal classifiers

We performed experiments with four state-of-the-art credal classifiers whose implementations are available in Weka-IP:[4] NCC, Lazy Naive Credal Classifier (LNCC), Credal Model Averaging (CMA), Credal Decision Tree (CDT). For a review of these methods, see [2]. We used the quadratic utility functions in Figure 2 to measure the sensitivity of the classifiers' performance to risk aversion. Tables 4 and 5 contain the median expected utility and median determinacy, respectively, of the classifiers over all datasets. From these tables, it is possible to see that CMA is the more determinate (in terms of median) of the classifiers, and also the one with highest expected utility. Likewise, NCC is the more indeterminate of the classifiers and it is also the one that benefits most from the increase of risk aversion: the increase in expected utility is greater for NCC than for other classifiers. CDT, on the other hand, seems to take little advantage on the increase of risk aversion.

$u_{0.5}$	NCC	LNCC	CMA	CDT
0.5	0.75	0.77	**0.81**	0.79
0.65	0.78	0.81	**0.82**	0.79
0.8	0.81	**0.83**	**0.83**	0.80

Table 4: Median expected utility of credal classifiers.

NCC	LNCC	CMA	CDT
95.03	97.90	99.56	97.49

Table 5: Median determinacy (in %) of credal classifiers on the datasets.

To evaluate whether there was a statistically significant difference between the performance of classifiers across datasets, we performed a Friedman test as suggested by Demsar [6]. The mean relative ranks according

[4] `http://decsai.ugr.es/~andrew/weka-ip.html`

to utility function appear in Table 6.[5] The tests found a significant difference under the risk-neutral utility function (i.e., discounted accuracy), but no significant difference under any of the risk-averse utility functions (u_{65} and u_{80}). We then applied a Nemenyi test to detect significant improvements between pairs of classifiers under the risk-neutral utility function. The result of the test is shown in Figure 6. Each segment represents an interval of significance for a classifier, with its center denoting the classifier's mean relative rank. A classifier X is significantly outperformed by a classifier Y if X's corresponding segment appears on the right of Y's segment and the two segments do not overlap. This is the case of NCC and CMA, and NCC and CDT, but not of any of the pairs NCC and LNCC, LNCC and CMA, LNCC and CDT, or CMA and CDT. In other words, NCC is significantly outperformed by CMA and CDT under the risk-neutral utility, but no other significant difference was observed among other pairs of classifiers.

$u_{0.5}$	NCC	LNCC	CMA	CDT
0.5	3.05	2.48	2.28	**2.18**
0.65	2.81	2.54	2.47	**2.18**
0.8	2.56	2.48	2.59	**2.36**

Table 6: Mean relative rank of credal classifiers on the datasets.

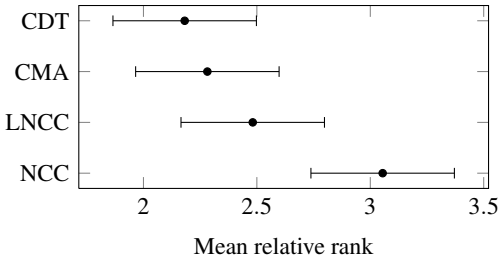


Figure 6: Nemenyi test for discounted accuracies of credal classifiers.

The determinacy and expected utility of the classifiers on each dataset are shown in Tables 12–15.

The squash-stored dataset provides an interesting case of the impact of risk aversion on the performance of credal classifiers. In this dataset, NCC is much more indeterminate than CDT (determinacy rate of 42.4% against 95.17%). Under linear utility (i.e., discounted accuracy), CDT outperforms the other classifiers (e.g., expected utility of 64.64% against an expected utility of 57.19% for NCC), but it is outperformed by NCC under u_{80}. In fact, NCC outperforms all the other classifiers in this setting, and obtains an expected utility of 72.6% against an expected utility of 65.95% for CDT. A similar phenomenon is seen in this dataset also for LNCC, which is even more indeterminate (determinacy of 35.97%) than NCC. LNCC's expected utility jumps from 49.84% (the lowest) under linear utility to 66.76% (the second highest) under u_{80}.

Since naive classifiers are sensitive to redundant information, we repeated the experiments after performing correlation-based feature selection. With feature selection, no classifier consistently outperformed another according to a Friedman test, irrespective of the utility function. Yet, CDT was found to be overall statistically significantly more determinate than others. Tables 7 and 8 show the median determinacy and median expected utility, respectively, of each classifier after the feature-selection step. By contrasting the results with those of Tables 5 and 4, one can see that NCC and LNCC become more determinate with feature selection, while CDT becomes more indeterminate (CMA's determinacy remains unaltered). Also, under the risk-neutral or the slightly risk-averse utility (i.e., for discounted accuracy or u_{65}), NCC and LNCC have

[5] For each dataset, we assign rank 1 to the classifier with highest expected utility, 2 to the classifier with the second highest expected utility, and so on. The mean relative rank is then the average of the ranks of a classifier over all datasets.

their median expected utility increased, while CDT exhibits a decrease in median expected utility and CMA is unaffected by the feature selection. Under u_{80}, feature selection produces a small decrease on median expected utilities of all classifiers. Relatively to others, NCC appears to benefit most from feature selection and risk aversion. This is also shown in Table 9, which reports the mean relative rankings of classifiers.

NCC	LNCC	CMA	CDT
96.33	99.14	99.56	98.13

Table 7: Median determinacy (in %) of credal classifiers after feature selection.

$u_{0.5}$	NCC	LNCC	CMA	CDT
0.5	0.79	**0.81**	**0.81**	0.77
0.65	0.79	0.81	**0.82**	0.78
0.8	0.80	0.82	**0.83**	0.78

Table 8: Median expected utility of credal classifiers after feature selection.

$u_{0.5}$	NCC	LNCC	CMA	CDT
0.5	2.89	2.41	**2.31**	2.39
0.65	2.69	2.44	2.45	**2.42**
0.8	2.50	2.50	**2.40**	2.60

Table 9: Mean relative rank of credal classifiers after feature selection.

All in all, among the four credal classifiers we tested, none consistently outperformed others when risk aversion is taken into account. Even under a risk-neutral utility, a simple correlation-based feature selection step was sufficient to boost NCC's performance and lead to no significance difference between classifiers.

9.3 Comparison with the non-deterministic classifier of Del Coz et al., 2009.

It is interesting to consider classifiers that return set-valued classifications and that have been developed outside the community of imprecise probability. In this section, we consider the *non-deterministic* classifier of [13] (NDC). This algorithm can be applied on top of any probabilistic classifier; it analyzes the posterior probability and decides whether to return a single class or a set of classes. Such algorithm thus works with a *single* posterior distribution, rather than with a posterior credal set. We implemented it to run on top of the NBC, so as to compare it against NCC: in this case, NCC and NDC can be seen as two alternatives for generating set-valued classifications with a naive classifier.

The NDC classifier optimizes the expected *F-measure* of the issued classification. The F-measure is a metric devised within the field of Information Retrieval and defined as the harmonic average of *recall* and *precision*. Let $\mathscr{K} \subseteq \mathscr{C}$ be a set-valued prediction made of k classes and $\mathbb{I}_{\mathscr{K}}$ the indicator function of $\mathscr{K}$, namely whether it contains or not the actual class. The recall (r) and the precision (p) are defined as:

$$r(\mathscr{K}) \coloneqq \mathbb{I}_{\mathscr{K}},$$
$$p(\mathscr{K}) \coloneqq \frac{\mathbb{I}_{\mathscr{K}}}{k}.$$

Thus, recall corresponds to whether or not a prediction is successful; precision corresponds to discounted accuracy. The F-measure F_β is defined as:

$$F_\beta(\mathscr{K}) := \frac{(1+\beta^2)pr}{\beta^2 p + r}. \tag{9}$$

The most common choices for β are $\beta = 1$ or $\beta = 2$, yielding respectively the F_1 and F_2 measures. In Table 10 we provide some examples comparing F_1 and F_2 measures with u_{65} and u_{80}.

Prediction	Precision (d. acc.)	Recall	F_1	F_2	u_{65}	u_{80}
1	1	1	1	1	1	1
$\{1,2\}$	0.5	1	0.67	0.83	0.65	0.80
$\{1,2,3\}$	0.33	1	0.50	0.71	0.47	0.60
$\{2,3,4\}$	0	0	0	0	0	0

Table 10: Scores of different predictions, assuming the actual class to be 1.

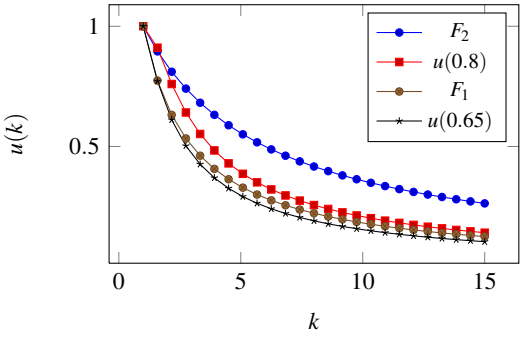


Figure 7: Comparison of F-metrics and quadratic utility functions.

A graphical comparison of the functions F_1, F_2, u_{65} and u_{80} is given in Figure 7; it shows that F_1 is comprised, for any value of k, between u_{65} and u_{80}, while F_2 rewards the indeterminate classifications much more than the other functions; this can be realized also from Table 10.

Thus the F-measure can be meaningfully interpreted, when dealing with indeterminate classifiers, as a utility function for set-valued classifications. In order to apply the NDC algorithm, the user should specify the value of β; we adopt $\beta := 1$, as F_1 is closer to both u_{65} and u_{80} than F_2.

A comparison between NCC and NDC (applied on top of NBC) has been already carried out in [13] considering UCI data sets and bioinformatics data sets, and reporting an overall superiority of NDC over NCC. It is somehow expected for NCC not to perform very well on raw bioinformatics data sets, which are characterized by several thousands of features and only some tenths of instances: they typically induce contingency tables with many zero counts, which make NCC excessively cautious; this issue is named the *feature problem* in [3].

Thus, we work with the collection of UCI data sets already considered in the previous sections. On each data set, we performed 10 runs of 10-folds cross-validation and eventually compared the performance of the two classifiers through a paired t-test. To complete the analysis, we then compared the performance of the two classifiers on the *whole* collection of data set through the Wilcoxon signed-rank test. All tests were performed with $\alpha := 0.05$. As utility functions we consider u_{65}, u_{80} and F_1.

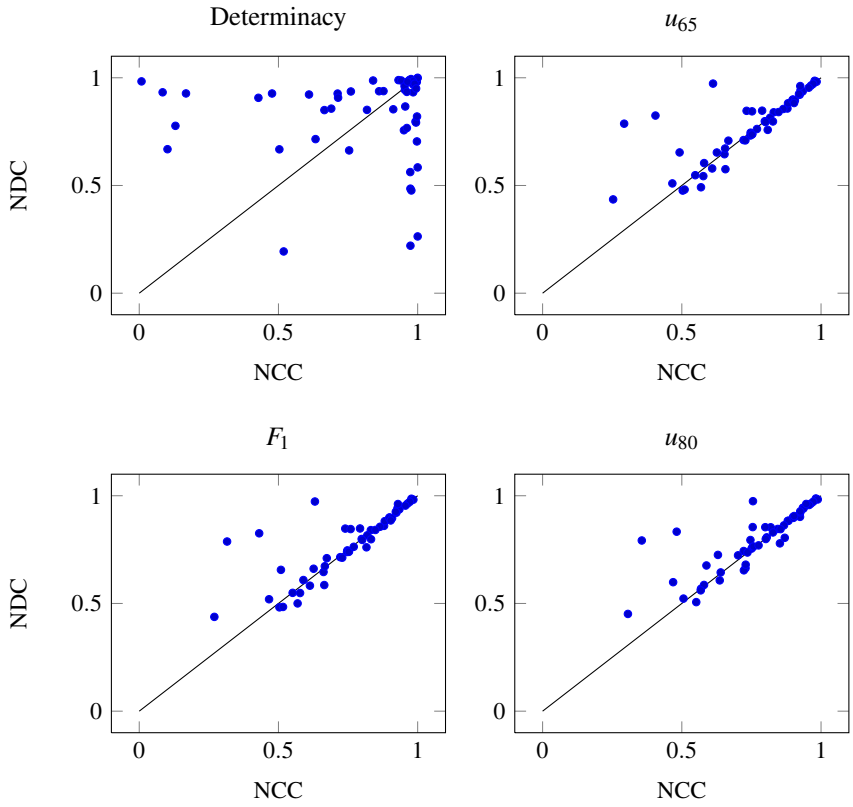


Figure 8: Scatter plots comparing NCC and NDC on 55 data sets.

Figure 8 compares NCC and NDC through scatter plots. The scatter plot of determinacy is very spread, showing that the two classifiers differently identify the instances over which to suspend the judgment. In fact, NCC is concerned with the sensitivity of the posterior distribution to the choice of the prior while NDC is concerned, grossly simplifying, with whether the probability of the most probable class is high enough. For instance NCC becomes more determinate as the sample size increases; this is not necessarily the case for NDC. Moreover, NCC becomes generally more indeterminate as the number of features increases [4]; on the contrary, NDC becomes *less* indeterminate in these cases, since an increased number of features emphasizes the 'probability overshooting' [11] phenomenon that characterizes naive classifiers.

The scatter plots of u_{65}, u_{80}, F_1 show a certain balance between the two classifiers. However, a bunch of data sets can be spotted, in which NCC is clearly outperformed; such data sets are indeed characterized by zero counts. Apart from those cases, the performance of the two classifiers appears quite balanced. This is also confirmed by the statistics of wins, ties and losses shown in Table 11. The performance of two classifiers on the *whole* collection of data sets is statistically different, in favor of NDC, only in the case of u_{80}. One might wonder why NDC, being designed to maximize F_1, outperforms NCC under u_{80} and not under F_1. As pointed out in [13], for a given $\tilde{\beta}$, NDC achieves the best performance under $F_{\tilde{\beta}}$ not necessarily using $\beta = \tilde{\beta}$ within its internal objective function.

Overall, NCC and NDC represent two very different approaches to yield indeterminate classifications. A nice characteristic of NDC is that it tunes its output according to the utility function being used; although in [13] this is done only for F_1 and F_2, with some further effort it should be possible to extend it also to more general utility functions. On the other hand, such an approach cannot identify prior-dependent classifications,

Utility	NCC wins	ties	NDC wins
u_{65}	21	14	18
F_1	20	16	17
u_{80}	14	13	26

Table 11: Comparison of NCC and NDC on the collection of data sets under various utilities.

which are clearly fragile. This is related to the discussion presented in the next section.

9.4 Utility-discounted accuracy vs. rejection option

The discussion in the previous section gives us the opportunity to comment on a related question, which is concerned with classifiers based on the *rejection option*. These are (usually) precise classifiers that do not classify an instance if the posterior probability of the optimal class is less than a fixed threshold t. Thus any determinate classifier that computes the posterior probabilities over the classes can easily be enabled to implement the rejection option. As we have mentioned in Section 9.3, also the NDC is close in spirit to a classifier based on the rejection option, although it cannot be just reduced to that.

To make things easier in the following discussion, we focus on a very simple setup. We consider a binary classification problem with $\mathscr{C} = \{1,2\}$; moreover, we assume that the class variable is independent of the feature variables, and that the population proportion (that is, the chance, or true probability) for class 1 equals 0.7. This means that the best one can hope for is to correctly classify the 70% of the instances.

Now assume that we implement the rejection option on the NBC, and that we select $t := 0.8$. This would turn the NBC into an indeterminate classifier, call it NBC$_t$, because we could identify the missing classification with the vacuous prediction $\{1,2\}$. How would this classifier compare with the NCC, under utility-discounted accuracy?

Assume that we evaluate the classifiers under u_{80}, and that the learning set is large enough so that after a while the NCC becomes nearly indistinguishable from the NBC. This means that, in those stationary conditions, the NCC will correctly predict 70% of the instances in the test set, and hence its expected utility will be 0.7 too. On the other hand, in the same conditions, the NBC$_t$ will constantly output the vacuous prediction $\{1,2\}$, simply because in the considered problem a class cannot have probability larger than or equal to 0.8. As a consequence, the expected utility of NBC$_t$ will be 0.8. The outcome of the comparison will then be in favor of NBC$_t$, even more so as the comparison will become more and more stable as the test set grows larger.

Even though the occurrence of a situation like the one described is fully consistent with the development we did of our utility-based metric, one could still be puzzled. In fact, on the one hand, it is correct that NBC$_t$ scores better than NCC in case your risk aversion is expressed by u_{80}: for you just prefer the vacuous prediction to a determinate prediction with probability 0.7. On the other hand, if our aim is to fairly compare classifiers, perhaps we should consider that the outcome of the comparison is not determined by a fault of NCC, but rather by the characteristics of the problem: in fact, NCC did learn from the data everything that could be learned.

More technically speaking, one should consider that the rejection option can be understood as a way to deal with cost-sensitive classification issues without introducing rewards other than 0-1 (as in matrix $\mathbb{I}$). In fact, decision theory prescribes that the optimal choice of a class under matrix $\mathbb{I}$ is the one that maximizes the posterior probability. Why should then one avoid selecting such a class, as in the case of the rejection option? The underlying motivation is that one has is mind a reward matrix different from $\mathbb{I}$: a matrix in which a wrong classification is penalized more than what happens using $\mathbb{I}$. For this reason, it is perceived that it is better not to issue a classication when the posterior probability of the class is not high enough.

This suggests that the classifiers based on the rejection option do not fully comply with the framework

that we have been developing, because a founding assumption of our framework is that $\mathbb{I}$ is the chosen reward matrix. Therefore, in order to make the comparison fair, it might be considered to exclude the classifiers that are based on some kind of rejection option.

Another, perhaps less preferrable, way could be that the classifiers based on the rejection option are only compared with one another. For instance, in the previous example problem, one could think of turning the NCC also into a classifier based on the rejection option: it would be sufficient to require it (in addition to its usual way of operating) to issue a vacuous classification whenever the maximum *lower* posterior probability over the classes does not exceed t.

Issues like the ones described could in fact have affected the comparison between NCC and NDC in the previous section: for instance, that NDC exhibits a remarkable improvement in the passage from u_{65} to u_{80}, could be an indication that some 'threshold' issue is at work there. As discussed above, a way to make a safer comparison between NCC and NDC, could be to enable NCC to group sets of classes based not only on its usual way of operating, but also according to the fact that the lower probability of a group of classes is high enough. Alternatively, this could be regarded as an extension of NDC to imprecision.

10 Conclusions

In this paper, we have tackled what we regard to be a very important, and conceptually involved, open problem: the empirical comparison of classifiers that issue set-valued (or indeterminate) predictions. We have proposed a new metric to this end, which has been derived in a principled way from a number of assumptions.

Our metric has shown itself to be based of two main, and opposing, components: the discounted accuracy, which represents a kind of objective performance of a classifier; and its variance, which represents the unreliability of the classifier, and whose contribution to the overall measure has to be weighted through subjective considerations of risk aversion.

We have given insights on the proposed metric, which we have called utility-discounted accuracy, and have used it to make extensive empirical comparisons of credal, as well as precise, classifiers. These tests, available in this form for the first time, show that the metric can easily be used in practice to finely compare classifiers, and to gain insights in their behaviour.

It would be important in the future to address the problem of evaluation of indeterminate predictors more in general. In fact, imprecise probability theories appear to have given so far very limited attention to the evaluation of indeterminate predictors. This is unfortunate, because indeterminacy is a fundamental characteristic of imprecise probability, and one that offers a privileged way to enhance reliability. Moreover, for any theory that claims itself to be statistical, it seems inescapable that a model's performance is measured—also—on data.

On the other hand, the extension of the results in this paper to more general settings might be challenging. A major issue concerns cost-sensitive classification, that is, classification problems where the reward matrix is different from $\mathbb{I}$. The crucial point is that using $\mathbb{I}$ leads to zero reward whatever wrong classification is made. When we allow different incorrect predictions to be assigned different rewards, we can no longer follow the argumentation that in Theorem 1 led us to derive discounted accuracy: in other words, it does not seem easy to obtain an objective measure of performance under rewards more general than 0 and 1. Since this is a key passage in the costruction of a utility-based metric, it could be difficult to define a metric in such a generalized setup. This problem affects classification, but it would affect even more an extension of this work to set-valued predictions of a continuous quantity, as it happens in regression, because 0-1 rewards are just too restrictive in that case. This says once more that a cost-sensitive generalization is needed. Can it be achieved? Some recent work by Seidenfeld, Schervish and Kadane [16] seems to indicate that there may be some fundamental limitation to evaluate indeterminate predictors in general. However, such a work is based on the concept of scoring rule, and this might leave the door open to alternative approaches. It is anyway a question that deserves careful consideration.

Finally, we would like to recall that the work in this paper has shown that there are relations between the evaluation of credal classifiers and finance. This seems to make this work somewhat close to recent research that explores the connections between utility and machine learning [10]. Deepening these relations might be fruitful in future research.

Acknowledgements

The research in this paper has been partially supported by the Swiss NSF grants nos. 200020_134759 / 1, 200020_137680 / 1, 200020-132252, by the Hasler foundation grant n. 10030.

References

[1] S.T. Buccola. Portfolio selection under exponential and quadratic utility. *Western Journal of Agricultural Economics*, 7(1):43–51, 1982.

[2] F. Coolen, T. Augustin, G. De Cooman, and M. Troffaes, editors. *Introduction to Imprecise Probability*. in press.

[3] G. Corani and A. Benavoli. Restricting the idm for classification. *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Theory and Methods*, pages 328–337, 2010.

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

[5] G. Corani and M. Zaffalon. Lazy naive credal classifier. In J. Pei, L. Getoor, and A. de Keijzer, editors, *First ACM SIGKDD Workshop on Knowledge Discovery from Uncertain Data*, pages 30–37. ACM, 2009.

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

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

[8] R. O. Duda and P. E. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.

[9] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley, 2001. 2nd edition.

[10] C. Friedman and S. Sandow. *Utility-Based Learning from Data*. Chapman & Hall/CRC, Boca Raton, Fl, 2011.

[11] D.J. Hand and K. Yu. Idiot's Bayes: not so stupid after all? *International Statistical Review*, 69(3):385–398, 2001.

[12] W. Hlawitschka. The empirical nature of Taylor-series approximations to expected utility. *The American Economic Review*, 84(3):713–719, 1994.

[13] J. Jose del Coz and A. Bahamonde. Learning nondeterministic classifiers. *Journal of Machine Learning Research*, 10:2273–2293, 2009.

[14] 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.

[15] D. G. Luenberger. *Investment Science*. Oxford University Press, New York, 1998.

[16] T. Seidenfeld, M. J. Schervish, and J. B. Kadane. Forecasting with imprecise probabilities. In F. Coolen, G. de Cooman, T. Fetz, and M. Oberguggenberger, editors, *ISIPTA '11: Proceedings of the Seventh International Symposium on Imprecise Probability: Theories and Applications*, pages 317–326, Innsbruck, Austria, 2011. SIPTA.

[17] J. Stoer and C. Witzgall, editors. *Convexity and Optimization in Finite Dimensions*. Springer-Verlag, Berlin, 1970.

[18] G. Tsoumakas and I. Vlahavas. Random k-label sets: an ensemble method for multilabel classification. In J. N. Kok, J. Koronacki, R. López de Mántaras, S. Matwin, D. Mladenic, and A. Skowron, editors, *Proceedings of ECML 2007, 18th European Conference on Machine Learning*, volume 4701 of *Lecture Notes in Computer Science*, pages 406–417. Springer, 2007.

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

[20] M. Zaffalon. A credal approach to naive classification. In G. de Cooman, F. G. Cozman, S. Moral, and P. Walley, editors, *ISIPTA '99: Proceedings of the First International Symposium on Imprecise Probabilities and Their Applications*, pages 405–414, Universiteit Gent, Belgium, 1999.

[21] M. Zaffalon. Statistical inference of the naive credal classifier. In G. de Cooman, T. L. Fine, and T. Seidenfeld, editors, *ISIPTA '01: Proceedings of the Second International Symposium on Imprecise Probabilities and Their Applications*, pages 384–393, The Netherlands, 2001. Shaker.

[22] M. Zaffalon. The naive credal classifier. *Journal of Statistical Planning and Inference*, 105(1):5–21, 2002.

A Detailed numerical comparisons

In this section we report the comparison of classifiers detailed data set by data set.

Dataset	NCC	LNCC	CMA	CDT
anneal	28.85	44.74	99.47	**99.90**
audiology	6.97	6.26	**95.59**	94.54
wisconsin-breast-cancer	99.83	**100.00**	**100.00**	98.48
cmc	97.49	**100.00**	93.20	93.22
connect-4	99.70	**100.00**	99.77	97.49
contact-lenses	65.00	63.83	96.00	**100.00**
credit	98.09	**99.32**	97.86	96.35
german-credit	95.80	**99.86**	86.97	88.79
pima-diabetes	99.21	**100.00**	99.99	96.67
ecoli	90.40	93.91	**100.00**	96.20
eucalyptus	79.51	95.14	**98.56**	95.25
glass	67.93	77.25	**99.95**	96.50
grub-damage	60.54	73.66	55.67	**74.62**
haberman	94.63	**100.00**	**100.00**	96.31
hayes-roth	51.88	51.88	**59.38**	**59.38**
cleveland-14-heart-diseases	15.61	47.63	**99.44**	98.09
hungarian-14-heart-diseases	75.36	86.52	98.17	**98.24**
hepatitis	94.83	**97.73**	94.47	96.44
hypothyroid	87.31	95.54	99.77	**99.95**
ionosphere	97.12	96.84	**100.00**	96.69
iris	97.60	98.40	**100.00**	98.60
kr-vs-kp	99.18	99.97	96.02	**99.98**
labor	85.87	88.93	93.50	**96.47**
letter	95.24	99.65	**100.00**	89.97
liver-disorders	**100.00**	**100.00**	**100.00**	**100.00**
lymphography	58.10	53.62	90.40	**96.65**
monks1	**100.00**	**100.00**	**100.00**	99.64
monks3	**100.00**	**100.00**	**100.00**	**100.00**
monks-2	97.16	**100.00**	**100.00**	87.12
mushroom	96.69	**100.00**	**100.00**	**100.00**
nursery	99.72	**100.00**	99.77	99.80
optdigits	98.12	99.21	**100.00**	90.08
page-blocks	97.47	99.38	**100.00**	99.50
pasture-production	63.00	61.17	**98.50**	96.33
pendigits	98.06	99.65	**100.00**	94.87
postoperative	48.11	69.78	99.44	**99.89**
primary-tumor	9.61	16.94	**88.17**	80.09
segment	95.64	96.14	**99.99**	98.11
solar-flare-C	85.45	97.90	97.50	**100.00**
solar-flare-m	69.85	89.45	93.31	**98.95**
solar-flare-X	92.67	99.29	95.39	**100.00**
sonar	95.58	99.18	**99.33**	94.91
soybean	93.15	92.28	**100.00**	98.34
spambase	99.64	99.81	**100.00**	98.29
spect	95.03	**99.81**	84.94	95.42
splice	98.80	**99.65**	99.56	97.88
squash-stored	42.40	35.97	83.93	**95.17**
squash-unstored	34.13	26.67	95.87	**99.40**
tae	97.40	**100.00**	42.62	99.73
vowel	74.79	93.75	**99.97**	93.66
waveform	99.38	99.96	**100.00**	96.72
white-clover	10.14	24.62	94.88	**98.55**
wine	96.57	91.03	**100.00**	98.93
yeast	97.15	99.14	**100.00**	97.28
zoo	81.82	80.14	99.60	**100.00**

Table 12: Determinacy (%) of credal classifiers. Boldface indicates highest determinacy on a dataset.

Dataset	NCC	LNCC	CMA	CDT
anneal	60.86	69.66	97.81	**99.57**
audiology	21.30	19.62	73.11	**78.87**
wisconsin-breast-cancer	97.15	96.11	**97.18**	95.41
cmc	50.03	**50.40**	50.03	48.84
connect-4	72.15	73.26	72.15	**76.80**
contact-lenses	76.44	76.31	**85.17**	83.50
credit	**86.12**	84.27	85.80	83.52
german-credit	**74.56**	73.79	72.88	68.26
pima-diabetes	75.27	**75.31**	75.25	74.01
ecoli	80.11	80.67	**80.89**	79.93
eucalyptus	53.05	58.54	55.87	**62.36**
glass	63.05	66.18	**71.74**	68.55
grub-damage	**46.33**	45.77	33.57	35.88
haberman	72.04	**73.39**	71.57	72.92
hayes-roth	**58.44**	**58.44**	**58.44**	**58.44**
cleveland-14-heart-diseases	32.02	54.42	**82.99**	75.58
hungarian-14-heart-diseases	70.77	76.92	**83.99**	78.11
hepatitis	83.87	83.80	**83.88**	79.76
hypothyroid	92.80	96.81	98.65	**99.35**
ionosphere	89.39	87.80	89.66	**89.83**
iris	92.93	92.57	93.27	**93.37**
kr-vs-kp	87.80	95.43	87.77	**99.49**
labor	88.83	**89.47**	88.48	83.90
letter	74.35	**86.45**	74.84	77.14
liver-disorders	**56.85**	**56.85**	**56.85**	**56.85**
lymphography	68.42	65.73	**81.10**	72.75
monks1	74.64	**89.29**	74.64	80.03
monks3	96.39	96.73	96.39	**98.92**
monks-2	62.22	**65.72**	**65.72**	65.39
mushroom	97.36	99.99	98.63	**100.00**
nursery	90.27	95.82	90.00	**96.28**
optdigits	92.13	**93.87**	92.22	77.16
page-blocks	93.32	95.60	93.84	**96.19**
pasture-production	75.38	71.93	**80.33**	72.89
pendigits	87.99	**94.27**	88.34	88.12
postoperative	50.91	60.61	70.83	**71.04**
primary-tumor	19.43	21.65	35.98	**38.15**
segment	91.69	92.75	92.52	**94.06**
solar-flare-C	81.47	87.11	88.69	**89.86**
solar-flare-m	75.15	84.55	87.81	**88.19**
solar-flare-X	91.58	94.92	95.57	**97.84**
sonar	76.27	75.66	**76.61**	73.39
soybean	91.63	90.58	**91.80**	91.78
spambase	89.88	**93.64**	89.87	91.57
spect	79.12	**82.51**	76.74	79.03
splice	95.43	64.82	**96.24**	92.70
squash-stored	57.19	49.84	58.95	**64.64**
squash-unstored	57.09	51.16	68.20	**76.67**
tae	46.32	46.72	38.05	**46.78**
vowel	57.79	65.89	62.94	**75.61**
waveform	79.97	**84.03**	79.96	73.71
white-clover	38.56	45.97	54.37	**62.03**
wine	97.71	93.14	**98.32**	91.94
yeast	57.49	**57.63**	57.54	57.01
zoo	88.92	85.12	**96.25**	95.92

Table 13: Expected utility (%) of credal classifiers under u_{50}. Boldface indicates best performance on a dataset.

Dataset	NCC	LNCC	CMA	CDT
anneal	71.03	77.53	97.89	**99.58**
audiology	27.43	25.43	73.58	**79.44**
wisconsin-breast-cancer	**97.18**	96.11	**97.18**	95.64
cmc	50.33	50.40	**50.83**	49.62
connect-4	72.19	73.26	72.18	**77.13**
contact-lenses	81.34	81.32	**85.77**	83.50
credit	**86.40**	84.37	86.12	84.07
german-credit	**75.19**	73.81	74.83	69.94
pima-diabetes	**75.39**	75.31	75.26	74.51
ecoli	81.20	**81.26**	80.89	80.30
eucalyptus	55.32	59.06	56.04	**62.84**
glass	66.64	68.61	**71.75**	68.91
grub-damage	**50.88**	48.99	37.66	38.53
haberman	72.84	73.39	71.57	**73.48**
hayes-roth	**65.66**	**65.66**	64.53	64.53
cleveland-14-heart-diseases	39.77	58.82	**83.07**	75.82
hungarian-14-heart-diseases	73.01	78.16	**84.24**	78.35
hepatitis	84.64	84.14	**84.71**	80.30
hypothyroid	94.63	97.43	98.68	**99.36**
ionosphere	89.82	88.27	89.66	**90.33**
iris	93.29	92.80	93.27	**93.58**
kr-vs-kp	87.92	95.43	88.37	**99.49**
labor	90.95	**91.13**	89.46	84.43
letter	74.73	**86.48**	74.84	77.68
liver-disorders	**56.85**	**56.85**	**56.85**	**56.85**
lymphography	73.47	71.18	**82.47**	73.18
monks1	74.64	**89.29**	74.64	80.09
monks3	96.39	96.73	96.39	**98.92**
monks-2	62.65	65.72	65.72	**67.33**
mushroom	97.86	99.99	98.63	**100.00**
nursery	90.31	95.82	90.04	**96.31**
optdigits	92.35	**93.96**	92.22	77.79
page-blocks	93.56	95.67	93.84	**96.25**
pasture-production	**80.80**	77.38	80.56	73.38
pendigits	88.20	**94.31**	88.34	88.48
postoperative	57.24	63.99	70.92	**71.05**
primary-tumor	24.72	26.22	36.54	**39.07**
segment	92.22	93.15	92.52	**94.21**
solar-flare-C	83.00	87.25	89.02	**89.86**
solar-flare-m	78.42	85.55	**88.70**	88.33
solar-flare-X	92.68	95.02	96.26	**97.84**
sonar	**76.94**	75.78	76.71	74.15
soybean	**92.48**	91.38	91.80	91.99
spambase	89.93	**93.67**	89.87	91.82
spect	79.87	**82.54**	79.00	79.71
splice	95.60	64.87	**96.30**	92.99
squash-stored	64.90	58.30	61.01	**65.30**
squash-unstored	65.76	60.93	68.77	**76.75**
tae	46.60	46.72	45.09	**46.80**
vowel	60.56	66.53	62.95	**76.16**
waveform	80.07	**84.04**	79.96	74.15
white-clover	48.56	54.16	55.14	**62.19**
wine	98.23	94.39	**98.32**	92.08
yeast	**57.75**	57.70	57.54	57.25
zoo	91.30	87.07	**96.31**	95.92

Table 14: Expected utility (%) of credal classifiers under u_{65}. Boldface indicates best performance on a dataset.

Dataset	NCC	LNCC	CMA	CDT
anneal	81.20	85.39	97.97	**99.59**
audiology	33.56	31.24	74.06	**80.01**
wisconsin-breast-cancer	**97.21**	96.11	97.18	95.86
cmc	50.63	50.40	**51.62**	50.41
connect-4	72.22	73.26	72.21	**77.46**
contact-lenses	86.23	86.33	**86.37**	83.50
credit	**86.69**	84.47	86.44	84.62
german-credit	75.82	73.83	**76.78**	71.62
pima-diabetes	**75.50**	75.31	75.26	75.01
ecoli	**82.29**	81.85	80.89	80.68
eucalyptus	57.60	59.58	56.20	**63.33**
glass	70.23	71.04	**71.76**	69.26
grub-damage	**55.42**	52.22	41.76	41.18
haberman	73.65	73.39	71.57	**74.03**
hayes-roth	**72.87**	**72.87**	70.62	70.62
cleveland-14-heart-diseases	47.53	63.22	**83.16**	76.07
hungarian-14-heart-diseases	75.25	79.40	**84.50**	78.59
hepatitis	85.42	84.48	**85.54**	80.83
hypothyroid	96.47	98.04	98.70	**99.36**
ionosphere	90.25	88.75	89.66	**90.82**
iris	93.65	93.04	93.27	**93.79**
kr-vs-kp	88.05	95.44	88.96	**99.49**
labor	**93.07**	92.79	90.43	84.96
letter	75.12	**86.50**	74.84	78.22
liver-disorders	**56.85**	**56.85**	**56.85**	**56.85**
lymphography	78.51	76.64	**83.83**	73.61
monks1	74.64	**89.29**	74.64	80.14
monks3	96.39	96.73	96.39	**98.92**
monks-2	63.08	65.72	65.72	**69.26**
mushroom	98.35	99.99	98.63	**100.00**
nursery	90.35	95.82	90.07	**96.33**
optdigits	92.58	**94.04**	92.22	78.42
page-blocks	93.80	95.73	93.84	**96.30**
pasture-production	**86.23**	82.83	80.78	73.87
pendigits	88.42	**94.35**	88.34	88.83
postoperative	63.58	67.38	71.00	**71.07**
primary-tumor	30.01	30.79	37.10	**39.99**
segment	92.76	93.56	92.52	**94.36**
solar-flare-C	84.54	87.40	89.34	**89.86**
solar-flare-m	81.70	86.55	**89.58**	88.48
solar-flare-X	93.78	95.13	96.95	**97.84**
sonar	**77.60**	75.91	76.81	74.91
soybean	**93.34**	92.18	91.80	92.20
spambase	89.98	**93.70**	89.87	92.08
spect	80.61	**82.56**	81.26	80.40
splice	95.76	64.92	**96.37**	93.27
squash-stored	**72.60**	66.76	63.07	65.95
squash-unstored	74.43	70.71	69.34	**76.83**
tae	46.88	46.72	**52.12**	46.82
vowel	63.33	67.17	62.95	**76.72**
waveform	80.16	**84.04**	79.96	74.59
white-clover	58.56	62.34	55.90	**62.36**
wine	**98.74**	95.63	98.32	92.22
yeast	**58.01**	57.77	57.54	57.49
zoo	93.68	89.02	**96.37**	95.92

Table 15: Expected utility (%) of credal classifiers under u_{80}. Boldface indicates best performance on a dataset.

Dataset	NCC	LNCC	CMA	CDT
anneal	63.71	95.45	99.47	**99.70**
audiology	37.81	42.52	**95.59**	94.46
wisconsin-breast-cancer	99.83	**100.00**	**100.00**	98.48
cmc	98.15	**100.00**	93.20	98.30
connect-4	99.98	**100.00**	99.77	99.78
contact-lenses	65.00	63.17	**96.00**	95.50
credit	98.35	**99.65**	97.86	96.09
german-credit	98.30	**100.00**	86.97	96.66
pima-diabetes	99.01	**100.00**	99.99	98.93
ecoli	90.40	93.91	**100.00**	96.20
eucalyptus	98.87	98.49	98.56	**98.89**
glass	70.85	80.52	**99.95**	95.95
grub-damage	56.58	67.98	55.67	**76.41**
haberman	94.83	**100.00**	**100.00**	96.45
hayes-roth	51.88	51.88	**59.38**	**59.38**
cleveland-14-heart-diseases	67.32	89.41	**99.44**	97.24
hungarian-14-heart-diseases	90.02	96.10	98.17	**98.58**
hepatitis	96.33	**99.09**	94.47	94.71
hypothyroid	99.72	**99.96**	99.77	99.94
ionosphere	96.29	98.94	**100.00**	97.01
iris	97.93	96.93	**100.00**	98.13
kr-vs-kp	99.80	**100.00**	96.02	99.97
labor	86.80	92.73	93.50	**95.90**
letter	95.92	99.69	**100.00**	89.77
liver-disorders	**100.00**	**100.00**	**100.00**	**100.00**
lymphography	80.79	81.86	90.40	**94.69**
monks1	**100.00**	**100.00**	**100.00**	99.75
monks3	**100.00**	**100.00**	**100.00**	**100.00**
monks-2	97.57	**100.00**	**100.00**	92.18
mushroom	99.91	**100.00**	**100.00**	**100.00**
nursery	99.77	**100.00**	99.77	99.80
optdigits	98.19	99.30	**100.00**	89.83
page-blocks	98.87	99.88	**100.00**	99.50
pasture-production	75.92	75.92	**98.50**	96.42
pendigits	98.28	99.72	**100.00**	94.93
postoperative	72.44	87.11	99.44	**99.89**
primary-tumor	22.77	37.81	**88.17**	80.05
segment	96.90	97.65	**99.99**	98.20
solar-flare-C	95.85	**99.66**	97.50	99.56
solar-flare-m	93.87	97.98	93.31	**98.88**
solar-flare-X	98.42	99.84	95.39	**100.00**
sonar	95.72	**99.42**	99.33	94.86
soybean	93.57	92.87	**100.00**	98.61
spambase	99.84	99.96	**100.00**	98.67
spect	95.42	**100.00**	84.94	97.25
splice	99.42	**99.85**	99.56	98.09
squash-stored	43.30	47.87	83.93	**93.27**
squash-unstored	61.27	63.57	95.87	**99.03**
tae	97.40	**100.00**	42.62	99.73
vowel	86.70	97.81	**99.97**	92.88
waveform	99.37	99.97	**100.00**	96.81
white-clover	49.29	65.52	94.88	**98.71**
wine	96.57	91.48	**100.00**	98.81
yeast	97.15	99.14	**100.00**	97.28
zoo	85.29	84.99	99.60	**100.00**

Table 16: Determinacy (in %) of credal classifiers after feature selection. Boldface indicates highest determinacy on a dataset.

Dataset	NCC	LNCC	CMA	CDT
anneal	79.74	95.47	**97.81**	97.71
audiology	47.00	48.48	73.11	**76.34**
wisconsin-breast-cancer	97.15	96.11	**97.18**	95.41
cmc	51.05	50.67	50.03	**51.61**
connect-4	70.01	69.38	**72.15**	70.45
contact-lenses	72.58	72.31	**85.17**	76.83
credit	**85.80**	85.30	**85.80**	84.39
german-credit	**73.42**	73.11	72.88	72.45
pima-diabetes	**75.60**	75.51	75.25	74.91
ecoli	80.10	80.67	**80.89**	79.93
eucalyptus	**57.69**	57.67	55.87	57.65
glass	62.92	66.62	**71.74**	66.96
grub-damage	44.34	**45.47**	33.57	37.74
haberman	72.10	**73.39**	71.57	72.96
hayes-roth	**58.44**	**58.44**	**58.44**	**58.44**
cleveland-14-heart-diseases	65.27	75.98	**82.99**	75.90
hungarian-14-heart-diseases	78.54	81.67	**83.99**	78.83
hepatitis	82.67	82.84	**83.88**	80.68
hypothyroid	97.69	97.19	**98.65**	97.73
ionosphere	90.37	89.87	89.66	**90.70**
iris	93.90	**93.93**	93.27	93.60
kr-vs-kp	92.40	93.45	87.77	**94.17**
labor	84.67	84.77	**88.48**	83.65
letter	74.55	**84.28**	74.84	76.99
liver-disorders	**56.85**	**56.85**	**56.85**	**56.85**
lymphography	75.39	75.15	**81.10**	74.59
monks1	**74.64**	**74.64**	**74.64**	74.31
monks3	96.39	96.52	96.39	**98.65**
monks-2	62.95	65.41	**65.72**	58.82
mushroom	98.57	98.62	98.63	**99.02**
nursery	90.02	94.87	90.00	**95.03**
optdigits	91.94	**93.53**	92.22	77.31
page-blocks	95.19	**96.55**	93.84	96.14
pasture-production	73.36	72.68	**80.33**	73.19
pendigits	87.89	**93.91**	88.34	88.13
postoperative	59.67	65.57	70.83	**71.04**
primary-tumor	26.12	29.48	35.98	**37.28**
segment	93.36	94.00	92.52	**94.25**
solar-flare-C	85.82	88.21	88.69	**89.01**
solar-flare-m	86.52	87.41	87.81	**88.66**
solar-flare-X	94.45	96.93	95.57	**97.84**
sonar	76.59	75.27	**76.61**	72.49
soybean	91.26	90.36	91.80	**92.45**
spambase	92.42	**93.37**	89.87	91.84
spect	79.39	**83.61**	76.74	81.16
splice	95.82	79.67	**96.24**	93.23
squash-stored	49.46	46.29	58.95	**60.65**
squash-unstored	66.65	65.86	68.20	**77.02**
tae	46.32	46.72	38.05	**46.78**
vowel	58.70	65.27	62.94	**68.50**
waveform	80.08	**84.01**	79.96	73.91
white-clover	49.69	55.22	54.37	**60.64**
wine	97.79	93.53	**98.32**	91.96
yeast	57.49	**57.63**	57.54	57.01
zoo	89.34	88.36	**96.25**	92.85

Table 17: Expected utility (in %) of credal classifiers under u_{50} after feature selection. Boldface indicates best performance on a dataset.

Dataset	NCC	LNCC	CMA	CDT
anneal	84.99	96.09	**97.89**	97.74
audiology	52.19	52.99	73.58	**76.92**
wisconsin-breast-cancer	**97.18**	96.11	**97.18**	95.64
cmc	51.29	50.67	50.83	**51.82**
connect-4	70.01	69.38	**72.18**	70.48
contact-lenses	77.31	77.04	**85.77**	77.35
credit	86.04	85.36	**86.12**	84.98
german-credit	73.67	73.11	**74.83**	72.95
pima-diabetes	**75.75**	75.51	75.26	75.07
ecoli	81.19	**81.26**	80.89	80.30
eucalyptus	57.83	**57.84**	56.04	57.78
glass	66.05	68.62	**71.75**	67.38
grub-damage	**49.42**	49.20	37.66	40.30
haberman	72.88	73.39	71.57	**73.49**
hayes-roth	**65.66**	**65.66**	64.53	64.53
cleveland-14-heart-diseases	67.92	76.85	**83.07**	76.30
hungarian-14-heart-diseases	79.34	82.03	**84.24**	79.04
hepatitis	83.22	82.97	**84.71**	81.48
hypothyroid	97.72	97.19	**98.68**	97.74
ionosphere	90.93	90.03	89.66	**91.15**
iris	94.21	**94.39**	93.27	93.88
kr-vs-kp	92.43	93.45	88.37	**94.18**
labor	86.65	85.86	**89.46**	84.26
letter	74.86	**84.30**	74.84	77.53
liver-disorders	**56.85**	**56.85**	**56.85**	**56.85**
lymphography	77.55	77.27	**82.47**	75.30
monks1	**74.64**	**74.64**	**74.64**	74.35
monks3	96.39	96.52	96.39	**98.65**
monks-2	63.31	65.41	**65.72**	59.99
mushroom	98.58	98.62	98.63	**99.02**
nursery	90.05	94.87	90.04	**95.06**
optdigits	92.16	**93.61**	92.22	77.95
page-blocks	95.31	**96.57**	93.84	96.20
pasture-production	76.83	76.08	**80.56**	73.70
pendigits	88.09	**93.94**	88.34	88.49
postoperative	62.93	66.98	70.92	**71.05**
primary-tumor	30.80	33.04	36.54	**38.34**
segment	93.75	94.24	92.52	**94.39**
solar-flare-C	86.30	88.24	89.02	**89.07**
solar-flare-m	87.11	87.55	88.70	**88.83**
solar-flare-X	94.69	96.95	96.26	**97.84**
sonar	**77.23**	75.35	76.71	73.26
soybean	92.08	91.11	91.80	**92.64**
spambase	92.45	**93.38**	89.87	92.04
spect	80.07	**83.61**	79.00	81.57
splice	95.91	79.69	**96.30**	93.48
squash-stored	57.07	53.33	61.01	**61.53**
squash-unstored	71.80	70.67	68.77	**77.16**
tae	46.60	46.72	45.09	**46.80**
vowel	60.08	65.46	62.95	**69.17**
waveform	80.17	**84.02**	79.96	74.34
white-clover	55.28	58.87	55.14	**60.80**
wine	98.30	94.71	**98.32**	92.13
yeast	**57.75**	57.70	57.54	57.25
zoo	91.24	90.06	**96.31**	92.85

Table 18: Expected utility (in %) of credal classifiers underu_{65} after feature selection. Boldface indicates best performance on a dataset.

Dataset	NCC	LNCC	CMA	CDT
anneal	90.25	96.71	**97.97**	97.77
audiology	57.39	57.50	74.06	**77.50**
wisconsin-breast-cancer	**97.21**	96.11	97.18	95.86
cmc	51.53	50.67	51.62	**52.02**
connect-4	70.02	69.38	**72.21**	70.50
contact-lenses	82.03	81.77	**86.37**	77.87
credit	86.29	85.41	**86.44**	85.57
german-credit	73.93	73.11	**76.78**	73.45
pima-diabetes	**75.89**	75.51	75.26	75.23
ecoli	**82.29**	81.85	80.89	80.68
eucalyptus	57.96	**58.00**	56.20	57.90
glass	69.18	70.63	**71.76**	67.79
grub-damage	**54.50**	52.93	41.76	42.86
haberman	73.65	73.39	71.57	**74.02**
hayes-roth	**72.87**	**72.87**	70.62	70.62
cleveland-14-heart-diseases	70.58	77.73	**83.16**	76.69
hungarian-14-heart-diseases	80.14	82.38	**84.50**	79.25
hepatitis	83.77	83.11	**85.54**	82.27
hypothyroid	97.75	97.19	**98.70**	97.75
ionosphere	91.49	90.19	89.66	**91.60**
iris	94.52	**94.85**	93.27	94.16
kr-vs-kp	92.46	93.45	88.96	**94.18**
labor	88.63	86.95	**90.43**	84.88
letter	75.18	**84.33**	74.84	78.08
liver-disorders	**56.85**	**56.85**	**56.85**	**56.85**
lymphography	79.71	79.38	**83.83**	76.01
monks1	**74.64**	**74.64**	**74.64**	74.39
monks3	96.39	96.52	96.39	**98.65**
monks-2	63.68	65.41	**65.72**	61.17
mushroom	98.60	98.62	98.63	**99.02**
nursery	90.08	94.87	90.07	**95.09**
optdigits	92.37	**93.69**	92.22	78.60
page-blocks	95.43	**96.58**	93.84	96.25
pasture-production	80.30	79.48	**80.78**	74.20
pendigits	88.28	**93.97**	88.34	88.84
postoperative	66.20	68.38	71.00	**71.07**
primary-tumor	35.48	36.61	37.10	**39.41**
segment	94.13	94.48	92.52	**94.54**
solar-flare-C	86.78	88.26	**89.34**	89.12
solar-flare-m	87.71	87.68	**89.58**	88.99
solar-flare-X	94.93	96.97	96.95	**97.84**
sonar	**77.88**	75.44	76.81	74.03
soybean	**92.89**	91.87	91.80	92.84
spambase	92.47	**93.38**	89.87	92.24
spect	80.76	**83.61**	81.26	81.98
splice	95.99	79.71	**96.37**	93.74
squash-stored	**64.68**	60.36	63.07	62.41
squash-unstored	76.96	75.47	69.34	**77.29**
tae	46.88	46.72	**52.12**	46.82
vowel	61.46	65.66	62.95	**69.85**
waveform	80.26	**84.02**	79.96	74.77
white-clover	60.87	**62.51**	55.90	60.96
wine	**98.81**	95.88	98.32	92.29
yeast	**58.01**	57.77	57.54	57.49
zoo	93.13	91.76	**96.37**	92.85

Table 19: Expected utility (in %) of credal classifiers under u_{80} after feature selection. Boldface indicates best performance on a dataset.