Robust Classification of Multivariate Time Series by Imprecise Hidden Markov Models[☆]

Alessandro Antonucci[1,*], Rocco de Rosa[2], Alessandro Giusti[1], Fabio Cuzzolin[3]

Abstract

A novel technique to classify time series with imprecise hidden Markov models is presented. The learning of these models is achieved by coupling the EM algorithm with the imprecise Dirichlet model. In the stationarity limit, each model corresponds to an imprecise mixture of Gaussian densities, this reducing the problem to the classification of static, imprecise-probabilistic, information. Two classifiers, one based on the expected value of the mixture, the other on the Bhattacharyya distance between pairs of mixtures, are developed. The computation of the bounds of these descriptors with respect to the imprecise quantification of the parameters is reduced to, respectively, linear and quadratic optimization tasks, and hence efficiently solved. Classification is performed by extending the k-nearest neighbors approach to interval-valued data. The classifiers are credal, this means that multiple class labels can be returned in the output. Experiments on benchmark datasets for computer vision show that these methods achieve the required robustness whilst outperforming other precise and imprecise methods.

Keywords: Multivariate time series, classification, credal classification, hidden Markov models, Markov chains, Gaussian mixtures, imprecise probability, credal sets, credal networks, Bhattacharyya distance.

[☆] This paper is a revised and extended version of [1].

[*] Corresponding author.

Email addresses: alessandro@idsia.ch (Alessandro Antonucci), rocco.derosa@unimi.it (Rocco de Rosa), alessandrog@idsia.ch (Alessandro Giusti), fabio.cuzzolin@brookes.ac.uk (Fabio Cuzzolin)

[1] Istituto Dalle Molle di Studi sull'Intelligenza Artificiale, Manno (Switzerland).

[2] Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano, Milano (Italy).

[3] Department of Computing, Oxford Brookes University Wheatley Campus, Wheatley, Oxford (United Kingdom).

1. Introduction

The theory of *imprecise probability* [2] extends the Bayesian theory of subjective probability to cope with sets of distributions. This potentially provides more robust and realistic models of uncertainty. These ideas have been applied to classification and a number of classifiers based on imprecise probabilities are already available. Most of these approaches are based on graphical models, whose parameters are imprecisely quantified with a set of priors by means of the *imprecise Dirichlet model* [3]. The first attempt in this direction is the *naive credal classifier* [4], which generalizes the naive Bayes classifier to imprecisely specified probabilities. Each prior in the imprecise Dirichlet model defines a precise classifier. When two precise classifiers of this kind assign a different class label to the same instance, the imprecise classifier returns both labels, and the instance is said to be *prior-dependent*. Conversely, when a single label is returned, this is independent of the prior. Classifiers of this kind, possibly returning multiple class labels in the output, are called *credal*.[4] The separation between prior-dependent and other instances induced by a credal classifier typically corresponds to a distinction between hard and easy-to-classify instances, with the accuracy of a precise classifier significantly lower on the prior-dependent instances rather than on the prior-independent ones. In this sense, credal classifiers are suitable as preprocessing tools, assigning the right class label to prior-independent instances and partially suspending the judgement otherwise.

Despite the relatively large number of credal classifiers proposed in the literature, no credal models specifically intended to classify temporal data have been developed so far.[5] This is cumbersome since, on the other side, dynamical models such as Markov chains and *hidden Markov models* (HMMs) have been already extended to imprecise probabilities to model non-stationary dynamic processes [9, 10]. As a matter of fact, HMMs in their precise formulation have been often applied to classification of time series (e.g., [11]), while no similar attempts have been made in the imprecise case. This can be partially explained by the lack of algorithms to learn imprecise-probabilistic models from incomplete data (e.g., be-

[4] Besides the naive, other examples of credal classifiers proposed in the literature include models with more complex topologies [5] and imprecise averages of precise models [6]. Alternative quantification techniques not based on the imprecise Dirichlet model have been proposed for general topologies in [7].

[5] The only exception is a previous work of the authors [8]. The algorithms in the present paper are a natural evolution of those approaches. Numerical tests showing much better performances of the new methods are reported in Section 7.

cause referred to hidden variables) and, more marginally, by the lack of suitable inference algorithms.

It therefore seems natural to merge these two lines of research and develop credal classifiers for time series based on imprecise HMMs. To achieve that, we first show how to learn an imprecise HMM from a discrete-time sequence. The technique, already tested in previous works [8, 12] combines the imprecise Dirichlet model with the popular EM algorithm, generally used to learn precise HMMs. After this step, each sequence is associated with an imprecise HMM. In the limit of infinitely long models, HMMs might converge to a condition of *stationarity*, even in the imprecise case [13, 14]. A major claim of this paper is that, in this limit, the model becomes considerably simpler without losing valuable information for classification purposes.

In the stationarity limit, the imprecise HMM becomes an *imprecise mixture* (i.e., with multiple specification of the weights) of Gaussian densities over the observable variable (i.e., the joint observation of the features). Two novel algorithms are proposed to perform classification with these models. The first, called IHMM-E, evaluates the mixture expected value, which becomes a static attribute for a standard classification setup. The second, called IHMM-B, uses the Bhattacharyya distance between two mixtures as a descriptor of the dissimilarity level between sequences. Being associated with imprecise-probabilistic models, those descriptors cannot be precisely evaluated and only their lower and upper bounds with respect to the constraints on the parameters can be evaluated. This is done efficiently by solving a linear (for IHMM-E) and a quadratic (for IHMM-B) optimization task.

After this step, IHMM-E summarizes the sequence as an interval-valued observation in the feature space. To classify this kind of information, a generalization of the *k-nearest neighbors* algorithm to support multivariate interval data is developed. The same approach can be used to process the interval-valued (univariate) distances between sequences returned by IHMM-B. Both algorithms are credal classifiers for time series, possibly assigning more than a single class label to a sequence. Performances are tested on some of the most important computer vision benchmarks. The methods we propose achieve the required robustness in the evaluation of the class labels to be assigned to a sequence and outperform alternative imprecise methods with respect to state-of-the-art metrics [15] to compare performances of credal and traditional classifiers.

The performance is also good when compared with *dynamic time warping*, the state-of-the-art approach to the classification of time series. The reason is the high dimensionality of the computer vision data: dynamic time warping is less effective when coping with multivariate data [16], while the methods in this paper are almost unaffected by the dimensionality of the features.

The paper is organized as follows. In Section 2, we introduce the basic ideas in the special case of precise HMMs obtained from univariate data. Then, in Section 3, we define imprecise HMMs and discuss the learning of these models from multivariate data. The new algorithms IHMM-E and IHMM-B are detailed in Sections 4 and 5. A summary of the two methods together with a discussion about their computational complexity and the performance evaluation are in Section 6. Experiments and conclusive remarks are in Sections 7 and 8.

2. Time Series Classification

Let us introduce the key features of our approach and the necessary formalism in the precise univariate case. Variables $O_1, O_2, \ldots, O_T$ denote the observations of a particular phenomenon at T different (discrete) times. These are assumed to be *observable*, i.e., their actual (real) values are available and denoted by $o_1, o_2, \ldots, o_T$. If the observations are all sampled from the same distribution, say $P(O)$, the empirical mean converges to its theoretical value (strong law of large numbers):

$$\lim_{T \to +\infty} \frac{\sum_{i=1}^{T} o_i}{T} = \int_{-\infty}^{+\infty} o \cdot P(o) \cdot \mathrm{d}o. \tag{1}$$

Under the stationarity assumption, the empirical mean is therefore a sensible descriptor of the sequence. More generally, observations at different times can be sampled from different distributions (i.e., the process can be non-stationary). Such a situation can be modeled by pairing O_t with an auxiliary discrete variable X_t, for each $t = 1, \ldots, T$. The values of variables $\{X_t\}_{t=1}^{T}$ are indexing the generating distributions: all these variables should therefore take values from the same set, say $\mathcal{X}$, whose M elements are in one-to-one correspondence with the different distributions. In other words, for each $t = 1, \ldots, T$, O_t is sampled from $P(O_t|X_t = x_t)$, and $P(O|x_{t'}) = P(O|x_{t''})$ if and only if $x_{t'} = x_{t''}$. Variables $\{X_t\}_{t=1}^{T}$ are assumed to be *hidden*, i.e., their actual values are not directly observable. The modeling of the generative process requires therefore the assessment of the joint mass function $P(X_1, \ldots, X_T)$. This becomes considerably simpler under the *Markovian assumption*: given X_{t-1}, all previous values of X are irrelevant to X_t, i.e., $P(X_t|x_{t-1}, x_{t-2}, \ldots, x_1) = P(X_t|x_{t-1})$. Together with the chain rule, this implies the factorization:

$$P(x_1, \ldots, x_T) := P(x_1) \cdot \prod_{t=2}^{T} P(x_t|x_{t-1}), \tag{2}$$

for each $(x_1, \ldots, x_T) \in \mathcal{X}^T$. If the transition probabilities among the hidden variables are independent of t, the specification of the joint model reduces to the assessment of $P(X_1)$ and $P(X_t|X_{t-1})$, i.e., $O(M^2)$ parameters. A model of this kind is called a time-homogeneous *Markov chain*. If all the transition probabilities are strictly positive, the model assumes a stationary behaviour on long sequences, i.e., the following limit exists:[6]

$$\tilde{P}(x) := \lim_{T\to\infty} P(X_T = x), \tag{3}$$

for each $x \in \mathcal{X}$, where the probability on the right-hand side is obtained by marginalizing out all the variables, apart from X_T, in the joint mass function in Equation (2). The marginal $\tilde{P}$ over $\mathcal{X}$ is called the *stationary mass function* of the Markov chain and it can be efficiently computed by standard techniques. In this limit, even the probability distribution over the observation becomes stationary:

$$\tilde{P}(o) = \sum_{x\in\mathcal{X}} P(o|x) \cdot \tilde{P}(x). \tag{4}$$

Again, as in Equation (1), the empirical mean converges to the theoretical value:

$$\lim_{T\to+\infty} \frac{\sum_{i=1}^{T} o_i}{T} = \sum_{x\in\mathcal{X}} \tilde{P}(x) \cdot \int_{-\infty}^{+\infty} o \cdot P(o|x) \cdot \mathrm{d}o. \tag{5}$$

The descriptor on the right-hand side of Equation (5) can be used as a static feature to be processed by standard classification algorithms. Yet, instead of considering only its mean, the whole distribution $\tilde{P}(O)$ might provide a more informative static feature to be used for classification.[7] This can be achieved by evaluating pairwise dissimilarity levels among the distributions associated to the different sequences. In particular, given two distributions $\tilde{P}(O)$ and $\tilde{Q}(O)$, we characterize their mutual dissimilarity in terms of the popular *Bhattacharyya distance*:

$$\delta_{\mathrm{Bh}}(\tilde{P}, \tilde{Q}) := -\ln \int_{-\infty}^{+\infty} \sqrt{\tilde{P}(o) \cdot \tilde{Q}(o)} \cdot \mathrm{d}o, \tag{6}$$

which evaluates the overlap between statistical samples generated by the two distributions. In the remainder of this paper we develop two algorithms based, respectively, on the weighted mean in Equation (5) and on the distance in Equation

[6] We refer the reader to the literature on Markov chains (e.g., [17, Chap. 12]) for a complete characterization of the existence of the limit in Equation (3).

[7] The variable X is an auxiliary hidden variable lacking a direct physical interpretation. It would therefore make no sense to consider also this variable in the comparison with other models.

(6). These ideas are extended to the imprecise-probabilistic framework, taking into account the case of multivariate observations. Overall, this leads to the specification of two credal classifier for time series described, respectively, in Sections 4 and 5. Before doing that, let us first formalize the notion of HMM in both the precise and the imprecise case.

3. Imprecise Hidden Markov Models

3.1. Definition

In this section we present imprecise HMMs as a generalization of standard HMMs. This is based on the fundamental notion of credal set, which is reviewed first. Following [18], a *credal set* over a categorical variable X is a closed and convex set $K(X)$ made of probability mass functions over X. We focus on finitely generated credal sets, this means that the set of the extreme points of $K(X)$, denoted as $\mathrm{ext}[K(X)]$, has finite cardinality.

A Markov chain defined as in the previous section can be easily extended to the imprecise framework by replacing probability mass functions with credal sets: $P(X_1)$ is replaced by $K(X_1)$ and $P(X_t|x_{t-1})$ by $K(X_t|x_{t-1})$ for each $x_{t-1} \in \mathcal{X}$. While a Markov chain defines a joint mass function as in Equation (2), an *imprecise Markov chain* defines a joint credal set $K(X_1, \ldots, X_T)$ made of (the convexification of) all the joint mass functions $P(X_1, \ldots, X_T)$ obtained as in Equation (2) with $P(X_1) \in K(X_1)$ and $P(X_t|x_{t-1}) \in K(X_t|x_{t-1})$, for each $x_{t-1} \in \mathcal{X}$.[8] Given the joint credal set $K(X_1, \ldots, X_T)$, a stationary credal set $\tilde{K}(X)$, analogous to the stationary mass function in Equation (3), can be obtained by marginalizing out all the variables apart from X_T and taking the limit $T \to \infty$. This is shown to exist if all the original credal sets assign strictly positive probability to any event. The limit behaviour of imprecise Markov chains has been studied in [13], where a formula to compute $\tilde{K}(X)$ has been derived.[9] This formula is reported in Appendix B.

As in the previous section, for each $t = 1, \ldots, T$, the categorical variable X_t is in correspondence with the continuous variable O_t. Given X_t, any other variable is assumed to be irrelevant to O_t; the conditional distributions $P(O_t|X_t)$ can be

[8] This joint credal set is also called the *strong extension* of the imprecise Markov chain. As shown by Proposition 1 in [19], this credal set can be obtained by considering only the extreme points of the credal sets and then taking the convex hull.

[9] The cited paper consider the so-called *epistemic extension* of an imprecise Markov chain, which in general corresponds to a larger credal set. Yet, marginal inferences based on the two models have been proved to be equivalent by [20].

therefore used to augment the Markov chain and define a (precise) HMM:

$$P(x_1,\ldots,x_T,o_1,\ldots,o_t) := P(x_1)\cdot P(o_1|x_1)\cdot\prod_{t=2}^{T}\left[P(x_t|x_{t-1})\cdot P(o_t|x_t)\right]. \tag{7}$$

Like the *transition* probabilities $P(X_t|X_{t-1})$, the *emission* terms $P(O_t|X_t)$ are also assumed time homogeneous (i.e., independent of t). Imprecise HMMs are similarly defined as the augmentation of an imprecise Markov chain.[10] Both precise and imprecise HMMs describe the generative process behind a temporal sequence of observations, corresponding to the variables $O_1,\ldots,O_T$. The discrete variables $X_1,\ldots,X_T$ are assumed to be hidden and, as outlined in the previous section, have the role of modeling the non-stationarity of the emission process.

3.2. Learning: Expectation Maximization + Imprecise Dirichlet Model

The variables $X_1,\ldots,X_T$ of a HMM, no matter whether precise or imprecise, are by definition assumed to be directly unobservable, i.e., *hidden*. An algorithm to learn the HMM parameters from incomplete data is therefore needed. In the precise case, the *expectation maximization* (EM) algorithm by [21] is a typical choice.

Given an initialization of the HMM parameters, the EM computes the probabilities of the different outcomes of the hidden variable. This is a probabilistic explanation of the values of the hidden variables in the dataset. *Expected counts* for the occurrences of these variables can be therefore estimated. These, generally speaking non-integer, values are used to re-estimate the model parameters. The procedure is iterated until convergence, which is known to take place when a local maximum of the likelihood is reached.

A similar approach can be considered in the imprecise case. However, the imprecise Dirichlet model, commonly used to learn credal sets, needs the data to be complete, and no alternatives for incomplete data are available.[11]

To bypass this problem it is sufficient to regard the expected counts returned by the EM as complete(d) data, and process them with the imprecise Dirichlet model.

[10] This is not the most general class of imprecise HMMs. Credal sets can also replace emission terms $P(O_t|X_t)$. Motivations to confine imprecision to the hidden layer are in Section 3.2.

[11] The conservative updating rule proposed by [22] can be regarded as a remarkable exception. The rule represents the most conservative approach to the modeling of the incompleteness process, and its application to this specific problem would produce overly imprecise results.

For the transition probabilities, this induces the following linear constraints:

$$\frac{E[n(x' \to x)]}{\sum_{x \in \mathcal{X}} E[n(x' \to x)] + s} \leq P(X_t = x | X_{t-1} = x') \leq \frac{E[n(x' \to x)] + s}{\sum_{x \in \mathcal{X}} E[n(x' \to x)] + s}, \tag{8}$$

where $E[n(x' \to x)]$ are the expected counts for consecutive pairs of hidden variables with values x' and x as computed by the EM after convergence. This corresponds to computing, for each $t = 2, \ldots, T$, the probability that $X_t = x$ and multiplying it for the probability that $X_{t-1} = x'$; the sum of these values over the whole sequence gives the expected count. Consequently, the sums in the denominators are marginal counts, i.e., $\sum_{x \in \mathcal{X}} E[n(x' \to x)] = E[n(x')]$. The nonnegative parameter s can be regarded as the *equivalent sample size* of the set of priors used by the imprecise Dirichlet model, thus affecting the level of imprecision of the model.

Considering the linear constraints in Equation (8) for each $x \in \mathcal{X}$, it is possible to compute the conditional credal set $K(X_t | X_{t-1} = x')$, which is intended as the credal set of probability mass functions over X_t consistent with those constraints. An expression analogous to Equation (8), with the marginal expected counts in the numerator and the total counts in the denominator is used to learn $K(X_1)$. Overall, this procedure defines an imprecise Markov chain over the hidden variables.

Regarding the number of states of the hidden variables $M := |\mathcal{X}|$, as already noticed in Footnote 6, these variables are lacking a direct interpretation. The value of M should be regarded as a parameter of the model, for which we typically adopt small values (e.g., $M = 3$ in our experiments in Section 7). The reason is that, with many categories, it is more likely to have at least a small marginal count. This makes large the difference between the upper and the lower bound in Equation (8), thus making the model quite imprecise.

Regarding the *emission* part of the model (i.e., the relation between hidden and observable variables), first note that the discussion was introduced in the case of a scalar observable O just for sake of simplicity. In many real-world problems, we often need to cope with sequences of arrays of $F > 1$ features, say $\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T$, with $\boldsymbol{o}_t \in \mathbb{R}^F$ for each $t = 1, \ldots, T$. To define a joint model over the features we assume their conditional independence given the corresponding hidden variable. A Gaussian distribution is indeed used, for each feature, to model the relation between hidden and observable variables:

$$P(\boldsymbol{o}_t | x_t) \cdot \mathrm{d}\boldsymbol{o}_t = \prod_{f=1}^{F} \mathcal{N}_{\sigma_f(x_t)}^{\mu_f(x_t)}(o_t^f) \cdot \mathrm{d}o_t^f, \tag{9}$$

where o_t^f is the f-th component of the array $\boldsymbol{o}_t$, $\mathcal{N}_\sigma^\mu$ is a Gaussian density with mean μ and standard deviation σ, and $\mu_f(x_t)$ and $\sigma_f(x_t)$ are the EM estimates for the mean and standard deviation of the Gaussian over O_t^f given that $X_t = x_t$.[12]

The clustering-based technique proposed in [23] defines a possible initialization of the emission terms in the EM, while uniform choices are adopted for the transition and the prior. This guarantees the strict positivity of the expected counts after convergence, which implies the strict positivity of the lower bound in Equation (8). The existence of the stationary credal set follows from this (see [24]). After this learning step, the sequence of observations in the F-dimensional space is associated with a time-homogeneous imprecise HMM, with transition and prior probabilities required to belong to credal sets and a precise (Gaussian) specification of the emission terms.

The overall procedure based on a generalization to imprecise probabilities of the classical EM algorithm for HMM should be regarded as an attempt to achieve more reliable, but imprecise, estimates in the HMM parameters. The choice of confining the imprecision to the hidden variables follows from the fact that the EM estimates for these variables are based on missing information, they appear therefore less reliable than those about the observable variables.

4. Using (Imprecise) Expected Values for Classification: IHMM-E

4.1. An Interval-Valued Descriptor for Imprecise HMMs

In this section we show how the descriptor proposed in Equation (5) for precise HMMs can be generalized to the case of the imprecise HMM which we learn from a sequence of feature vectors by means of the procedure described in Section 3.2. In the imprecise case the stationary mass function of a Markov chain is replaced by a *stationary credal set*, say $\tilde{K}(X)$. As shown in [13], its computation, which is briefly summarized in Appendix B, can be efficiently achieved by Choquet integration.[13]

Thus, in this generalized setup, distribution $\tilde{P}(X)$ in Equation (5) is only required to belong to $\tilde{K}(X)$. The procedure described in [25] is used to obtain an

[12] The choice of using a single Gaussian, separately for each feature, is just for the sake of simplicity. An extension of the methods proposed in this paper to a single multivariate Gaussian with non-diagonal covariance matrix would be straightforward, even with mixtures.

[13] A direct approximation of the stationary credal set can be based on the expected marginal counts $\{E[n(x)]\}_{x\in\mathcal{X}}$. This produces more precise credal sets because of the higher values of the counts, but, apart from the case of very long sequences, less accurate estimates.

outer approximation of $\tilde{K}(X)$ based on a finite number of linear constraints. Regarding the emission terms, nothing changes as they are assumed to be precise. Thus, for each feature o_f, with $f = 1, \ldots, F$, we evaluate the bounds of the expectation as

$$\underline{o}^f := \min_{\tilde{P}(X)\in\tilde{K}(X)} \sum_{x\in\mathcal{X}} \tilde{P}(x) \cdot \mu_f(x), \tag{10}$$

$$\overline{o}^f := \max_{\tilde{P}(X)\in\tilde{K}(X)} \sum_{x\in\mathcal{X}} \tilde{P}(x) \cdot \mu_f(x). \tag{11}$$

Both $\underline{o}^f$ and $\overline{o}^f$ are solutions of linear programs with $|\mathcal{X}|$ optimization variables and an equal number of linear constraints (see Appendix B). The interval $[\underline{o}^f, \overline{o}^f]$ represents therefore the range of the descriptor in Equation (5) associated to O^f in the case of imprecise HMMs.

The lower and upper vectors $\underline{\boldsymbol{o}}, \overline{\boldsymbol{o}} \in \mathbb{R}^F$ are indeed obtained by applying the optimization in Equations (10) and (11) to each feature. They define a hyperbox in the feature space, which can be regarded as the range of the F-dimensional version of the descriptor in Equation (5) when imprecise probabilities are introduced in the model. Overall, a static interval-valued summary of the information contained in the temporal sequence has been obtained: the sequence, which is a trajectory in the feature space is described by a hyperbox in the same space (Figure 1). In the next section, a standard approach to the classification of static data is extended to the case of interval data like the ones produced by this method.

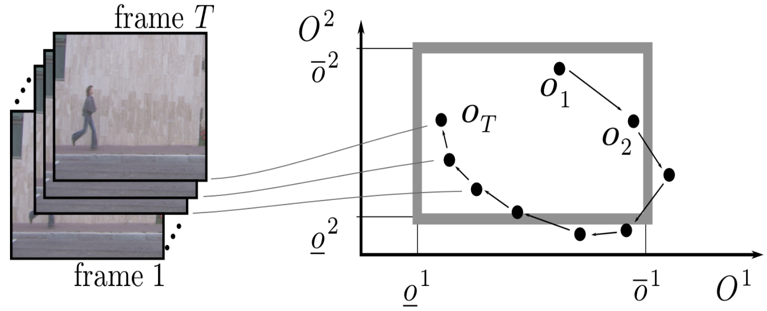

Figure 1: From trajectories to hyperboxes in the feature space. The example refers to footage data from which two features are extracted at the frame level.

4.2. Distances Between Hyperboxes

Consider the F-dimensional real space $\mathbb{R}^F$. Let us make it a metric space by considering, for instance, the *Manhattan* distance which, given $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^F$, defines the distance between them δ as

$$\delta(\boldsymbol{x}, \boldsymbol{y}) := \sum_{f=1}^{F} |x_f - y_f|. \tag{12}$$

Given two points $\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}} \in \mathbb{R}^F$ such that, for each $f = 1, \ldots, F$, $\underline{x}_f \leq \overline{x}_f$, the *hyperbox* associated with these two points is denoted by $[\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}]$ and defined as

$$[\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}] := \left\{ \boldsymbol{x} \in \mathbb{R}^F \,\middle|\, \underline{x}_f \leq x_f \leq \overline{x}_f \right\}. \tag{13}$$

The problem of extending a distance defined over points to hyperboxes can be solved by considering the ideas proposed in [26].

Given two hyperboxes, their distance can be characterized by means of a real interval whose bounds are, respectively, the minimum and the maximum distance (according to the distance defined for points) between every possible pair of elements in the two hyperboxes. Accordingly, the lower distance between two hyperboxes is:

$$\underline{\delta}([\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}], [\underline{\boldsymbol{y}}, \overline{\boldsymbol{y}}]) := \min_{\boldsymbol{x} \in [\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}], \boldsymbol{y} \in [\underline{\boldsymbol{y}}, \overline{\boldsymbol{y}}]} \delta(\boldsymbol{x}, \boldsymbol{y}), \tag{14}$$

and similarly, with the maximum instead of the minimum for the upper distance $\overline{\delta}([\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}], [\underline{\boldsymbol{y}}, \overline{\boldsymbol{y}}])$.[14] With the Manhattan distance in Equation (12), the evaluation of the lower (and similarly for the upper) distance as in Equation (14) takes a particularly simple form:

$$\underline{\delta}([\underline{\boldsymbol{x}}, \overline{\boldsymbol{x}}], [\underline{\boldsymbol{y}}, \overline{\boldsymbol{y}}]) = \sum_{f=1}^{F} \min_{\substack{\underline{x}_f \leq x_f \leq \overline{x}_f, \\ \underline{y}_f \leq y_f \leq \overline{y}_f}} |x_f - y_f|. \tag{15}$$

The optimization in the F-dimensional space is in fact reduced to F, independent, optimizations on the one-dimensional real space. Each task can be reduced to linear program whose optimum is in a combination of the extremes, unless intervals

[14] The one-sided Hausdorff distance from a hyperbox to the other is obtained by replacing the minimization over one of the two hyperboxes in Equation (14) with a maximization. The two Hausdorff distances are therefore between the lower and upper distances we propose. Although possibly leading to more informative classifications and close to the current approach, using the Haudorff distances here would reflect a hierarchical information about the points in the hyperboxes. This is not justified in the imprecise-probabilistic framework [27].

overlap. In other words:

$$\min_{\substack{\underline{x}_f \leq x_f \leq \overline{x}_f \\ \underline{y}_f \leq y_f \leq \overline{y}_f}} |x_f - y_f| = \min \left\{ \begin{array}{c} |\underline{x}_f - \underline{y}_f|, |\overline{x}_f - \underline{y}_f|, \\ |\underline{x}_f - \overline{y}_f|, |\overline{x}_f - \overline{y}_f| \end{array} \right\}, \tag{16}$$

unless $\overline{x}_f \geq \underline{y}_f$ or $\overline{y}_f \geq \underline{x}_f$, a case where the lower distance is clearly zero. A dual relation holds for the upper distance case with no special discussion in case of overlapping.

Replacing the Manhattan with the Euclidean distance makes little difference if we consider only the sum of the squared differences of the coordinates without the square root.[15] In this case the lower distance is the sum, for $f = 1, \ldots, F$ of the following terms:

$$\min_{\substack{\underline{x}_f \leq x_f \leq \overline{x}_f, \\ \underline{y}_f \leq y_f \leq \overline{y}_f}} (x_f - y_f)^2. \tag{17}$$

This is the minimum of a convex function, which is attained on the border of its (rectangular) domain. It is straightforward to check that the minimum should lie on one of the four extreme points of the domain. Thus, the minimum in Equation (17) is the minimum of the squares of the four quantities in Equation (16). Again, the only exception is when the two intervals overlap (the global minimum is in $x_f = y_f$), and the lower distance becomes zero. Similar considerations hold for the upper distance.

4.3. k-Nearest Neighbors Classification of Interval Data

The above defined interval-valued distance for hyperboxes is the key to extending the *k-nearest neighbors* (*k*-NN) algorithm to the case of interval-valued data. First, let us review the algorithm for pointwise data.

Let C denote a *class* variable taking its values in a finite set $\mathcal{C}$. Given a collection of supervised data $\{c^d, \boldsymbol{x}^d\}_{d=1}^D$ classification is intended as the problem of assigning a class label $\tilde{c} \in \mathcal{C}$ to a new instance $\tilde{\boldsymbol{x}}$ on the basis of the data. The *k*-NN algorithm for $k = 1$ assigns to $\tilde{\boldsymbol{x}}$ the label associated with the nearest instance, i.e., the assigned label is $\tilde{c} := c^{d^*}$ with

$$d^* = \mathrm{argmin}_{d=1,\ldots,D}\, \delta(\tilde{\boldsymbol{x}}, \boldsymbol{x}^d). \tag{18}$$

[15] The square root is a monotone function, which has no effect on the ranking-based classification method we define here.

For $k > 1$, the k nearest instances need to be considered instead: a voting procedure among the relative classes decides the label of the test instance.

To extend this approach to interval data just replace the sharp distance among points used in Equation (18) with the interval-valued distance for hyperboxes proposed in Section 4.2. However, to compare intervals instead of points a decision criterion is required.

To see that, consider for instance three hyperboxes and the two intervals describing the distance between the first hyperbox and, respectively, the second and the third. If the two intervals do not overlap, we can trivially identify which is the hyperbox nearer to the first one. Yet, in case of overlapping, this decision might be debatable. The most cautious approach is *interval dominance*, which simply suspends any decision in this case.

When applied to classification, interval dominance produces therefore a *credal* classifier, which might return more than a single class in the output. If the set of optimal classes according to this criterion is defined as C^*, we have that $c \in C^*$ if and only if there exists an instance $(c^i, \boldsymbol{x}^i)$ with $c^i = c$ such that there is no $d = 1, \ldots, n$, with $c^d \neq c$, satisfying the following dominance test:

$$\overline{\delta}([\underline{\boldsymbol{x}}^d, \overline{\boldsymbol{x}}^d], [\underline{\tilde{\boldsymbol{x}}}, \overline{\tilde{\boldsymbol{x}}}]) < \underline{\delta}([\underline{\boldsymbol{x}}^i, \overline{\boldsymbol{x}}^i], [\underline{\tilde{\boldsymbol{x}}}, \overline{\tilde{\boldsymbol{x}}}]). \tag{19}$$

Classes in the above defined set are said to be *undominated* because they correspond to instances in the dataset whose interval-valued distance from the test instance is not clearly bigger that the interval distance associated with any other instance. A demonstrative example is in Figure 2.

This approach can be extended to the case $k > 1$. To do that, consider the instances $\boldsymbol{x}^i$ such that no more than $k-1$ instances $\boldsymbol{x}^d$ satisfy the dominance test in Equation (19). Instances of this kind are among the k nearest ones to $\tilde{\boldsymbol{x}}$. A set of k instances can be extracted from this set and the voting procedure applied. The set of undominated classes C^* is the union of the labels obtained by iterating this procedure over all the possible extractions. Yet, for increasing values of k, the procedure becomes more demanding from a computational point of view (being roughly exponential in k) and the output very imprecise. We therefore suggest to keep the value $k = 1$, for which the procedure only takes linear time and the number of classes in output is minimal. This is also consistent with the typical choice of k in the classification of time series [28].

4.4. The IHMM-E Credal Classifier

By merging the results in Sections 3, 4.1, 4.2 and 4.3 we have a classifier for time series, to be called IHMM-E, based on imprecise HMMs. In summary, for

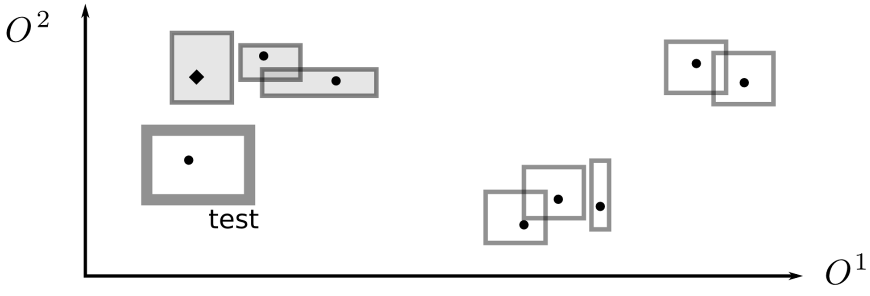

Figure 2: Rectangular data processed by the 1-NN classifier. Gray background denotes data whose interval distance from the test instance is undominated. Points inside the rectangles describe consistent precise data and the diamond is the nearest instance.

each sequence we: (i) learn an imprecise HMM (Section 3.2); (ii) compute its stationary credal set (Appendix B); (iii) solve the LP tasks required to compute the hyperbox associated with the sequence (Section 4.1). These supervised hyperboxes are finally used to perform k-NN credal classification (Section 4.3) based on the interval-valued distance between hyperboxes (Section 4.2).

5. Directly Coping with the Distributions: IHMM-B

The key idea of our approach so far is that considering the HMM in the limit of stationarity makes the model considerably simpler (and this is crucial for the extension to imprecise probabilities), and that classification does not suffer from this simplification. Moving to the stationarity limit basically transforms a dynamic model into a static model described by the limit distribution (or credal set) associated with the model. In this perspective, the choice of summarizing the static model by means of the expected value of the limit distribution as in Equation (5), which in the imprecise multivariate case generalizes to the expressions in Equation (10) and (11), is just one of the possible options.

Another approach might consist in coping directly with the limit distribution, which in the precise multivariate case is a mixture of Gaussians $\tilde{P}(\boldsymbol{O})$ such that:

$$\tilde{P}(\boldsymbol{o}) := \sum_{x\in\mathcal{X}} \tilde{P}(x) \cdot \prod_{f=1}^{F} \mathcal{N}_{\sigma^f(x)}^{\mu^f(x)}(o^f), \tag{20}$$

for each $\boldsymbol{o} \in \mathbb{R}^F$. As already noted in Section 2, classification can be based on the dissimilarity level between limit distributions associated to different models.

This can be identified with the Bhattacharyya distance, defined as in Equation (6) (in the multivariate case the integral should be over $\mathbb{R}^F$). In general, such a distance cannot be computed analytically, but a good approximation has been suggested in [29]. To show how the approximation works, consider a second HMM, with hidden variables taking values in $\mathcal{X}'$ and emission terms with mean $\mu^f(x')$ and variance $\sigma^f(x')$ for each $f = 1, \ldots, F$ and $x' \in \mathcal{X}'$. Let also $\tilde{Q}(X')$ denote the stationary mass function for this HMM. We call Bhattacharyya error, the following integral:

$$\epsilon_{\mathrm{Bh}}(\tilde{P}, \tilde{Q}) := \int_{\boldsymbol{o} \in \mathbb{R}^F} \sqrt{\tilde{P}(\boldsymbol{o}) \cdot \tilde{Q}(\boldsymbol{o})} \cdot \mathrm{d}\boldsymbol{o}, \tag{21}$$

whose relation with the Bhattacharyya distance is $\delta_{\mathrm{Bh}}(\tilde{P}, \tilde{Q}) = -\ln \epsilon_{\mathrm{Bh}}(\tilde{P}, \tilde{Q})$. The approximation for mixtures of Gaussians consists in taking the convexity bound:

$$\epsilon'_{\mathrm{Bh}}(\tilde{P}, \tilde{Q}) := \sqrt{\sum_{x \in \mathcal{X}, x' \in \mathcal{X}'} \tilde{P}(x) \cdot \tilde{Q}(x') \cdot \prod_{f=1}^{F} \epsilon^2_{\mathrm{Bh}}(\mathcal{N}^{\mu^f(x)}_{\sigma^f(x)}, \mathcal{N}^{\mu^f(x')}_{\sigma^f(x')})}, \tag{22}$$

where the Bhattacharyya errors and distances between two Gaussians can be computed analytically by means of the following formula [30]:

$$\delta_{\mathrm{Bh}}(\mathcal{N}^{\mu}_{\sigma}, \mathcal{N}^{\mu'}_{\sigma'}) := \frac{1}{4} \frac{(\mu - \mu')^2}{\sigma^2 + \sigma'^2} + \frac{1}{4} \ln \left[\frac{1}{4} \left(\frac{\sigma'^2}{\sigma^2} + \frac{\sigma^2}{\sigma'^2} + 2 \right) \right]. \tag{23}$$

In summary, in the precise case, classification can be performed by evaluating the distances between the limit distribution of the test instance and those of the training instances. As in Equation (18), the class label assigned to the test instance is the one belonging to the training instance at minimum distance. This method can be easily extended to the imprecise-probabilistic case. In this case, instead of a limit distribution defined as a mixture of Gaussians, we have a set of mixtures, also called a *credal mixture*, one for each $\tilde{P}(X) \in \tilde{K}(X)$. Following an approach similar to that in Section 4.2, we can simply characterize the level of dissimilarity between two credal mixtures by means of an interval, whose bounds are respectively the minimum and the maximum Bhattacharyya distance between mixtures of Gaussians with weights consistent with the stationary credal set of the corresponding models. This corresponds to optimizing the Bhattacharyya error in Equation (22), with respect to the optimization variables $\{\tilde{P}(x)\}_{x \in \mathcal{X}}$ and $\{\tilde{Q}(x')\}_{x' \in \mathcal{X}'}$. Considering that the square root is a monotone function and that the credal sets are defined by linear constraints, this is equivalent to a linearly constrained quadratic optimization task, with the objective function having form:

$$f(x,x') := \sum_{x\in\mathcal{X},x'\in\mathcal{X}'} k(x,x')\cdot\tilde{P}(x)\cdot\tilde{Q}(x') \tag{24}$$

with linear constraints $\tilde{P}(X) \in \tilde{K}(X)$ and $\tilde{Q}(X) \in \tilde{K}(X')$, and the nonnegative coefficient $k(x,x')$ being the product of the Bhattacharyya errors in Equation (22). In Appendix A we prove that both the minimization and the maximization of this problem can be efficiently solved. Note that the maximization corresponds to the upper bound of the Bhattacharyya error, which, because of monotonicity, defines the lower bound of the Bhattacharyya distance (and similarly for the minimization). Exactly as in Section 4.3, these interval-valued distances can be partially ranked by means of the interval dominance criterion in Equation (19) and hence used to perform credal classification.[16] We call this credal classifier IHMM-B.

6. Related Works, Complexity, and Performance Evaluation

Another credal classifier for time series based on imprecise HMMs, and called here IHMM-L, has been proposed in [8]. Each imprecise HMM learned from a supervised sequence is used to “explain” the test instance, i.e., the lower and upper bounds of the probability of the sequence are evaluated. These (probability) intervals are compared and the optimal classes according to interval dominance are eventually returned.

Regarding traditional (i.e., not based on imprecise probabilities) classifiers, *dynamic time warping* (DTW) is a popular state-of-the-art approach. Yet, its performance degrades in the multivariate case, i.e., $F > 1$ [16]. Our new classifiers are tested against IHMM-L and DTW in the next section.

Other approaches to the specific problem of classifying interval data have been also proposed. E.g., remaining in the imprecise-probabilistic area, the approach proposed in [31] can be used to define a support vector machine (SVM) for interval data. Yet, time complexity increases exponentially with the number of features, thus preventing an application of the method to data with high feature

[16] Numerical problems can be encountered because of the coefficient in the objective function in Equation (24), which are too small. These are fixed by dividing the objective function for the smallest Bhattacharyya error between a Gaussian term of the test instance and those of the training instances.

dimensionality. This is not the case for IHMM-E and IHMM-B, whose complexity is analyzed below.

6.1. Complexity Analysis

Let us first evaluate IHMM-E. Our approach to the learning of imprecise HMMs has the same time complexity as the precise case, namely $O(M^2TF)$. The computation of the stationary credal set is $O(T)$, while to evaluate the hyperboxes a LP task should be solved for each feature, i.e., roughly, $O(M^3F)$. Also the distance between two hyperboxes can be computed efficiently: the number of operations required is roughly four times the number of operations required to compute the distance between two points, both for Manhattan and Euclidean metrics. To classify a single instance as in Equation (19), lower and upper distances should be evaluated for all the sequences, i.e., $O(DF)$. Overall, the complexity is linear in the number of features and in the length of the sequence and polynomial in the number of hidden states. Similar results can be found also for space.

Regarding IHMM-B, everything is just the same, apart from the evaluation of the hyperboxes, which is replaced by the evaluation of the distances. This is a (single) quadratic optimization task. To specify the objective function $O(M^2F)$ time is required, while the solution of the problem is roughly cubic in the number of constraints, thus $O(M^3)$. The same conclusions for the previous algorithm are therefore valid also in this case.

6.2. Metrics for Credal Classifiers

Credal classifiers might return multiple classes in the output. Evaluating their performance requires therefore specific metrics, which are reviewed here. First, a characterization of the level of indeterminacy is achieved by: the *determinacy*, i.e., percentage of instances classified with a single label; the *average output size*, i.e., the average number of classes on instances for which multiple labels are returned. We normalize this number by dividing it by the total number of classes.

For accuracy we distinguish between: *single accuracy*, i.e., accuracy over instances classified with a single label; and *set accuracy*, i.e., the accuracy over the instances classified with multiple labels. In the latter case, classification is considered correct if the set of labels includes the true class.

A utility-based measure has been recently proposed in [15] to compare credal and precise classifiers with a single indicator. In our view, this is the most principled approach to compare the 0-1 loss of a traditional classifier with a utility score defined for credal classifiers. The starting point is the *discounted accuracy*, which rewards a prediction containing q classes with $1/q$ if it contains the true class, and with 0 otherwise. This indicator can be already compared to the accuracy achieved by a determinate classifier.

Yet, risk aversion demands higher utilities for indeterminate-but-correct outputs when compared with wrong-but-determinate ones [15]. Discounted accuracy is therefore modified by a (monotone) transformation u_w with $w \in [.65, .80]$. A conservative approach consists in evaluating the whole interval $[u_{.65}, u_{.80}]$ for each credal classifier and compare it with the (single-valued) accuracy of traditional classifiers. Interval dominance can be used indeed to partially rank performances.

We call *precise counterpart* of a credal classifier a classifier always returning a single class included in the output of the credal classifier. As an example, both IHMM-E and IHMM-B admit a precise counterpart based on a precise HMM, which corresponds to set $s = 0$ in the imprecise Dirichlet model. If a precise counterpart is defined, it is also possible to evaluate: the *precise single accuracy*, i.e., the accuracy of the precise classifier when the credal returns a single label; and the *precise set accuracy*, i.e., the accuracy of the precise classifier when the credal returns multiple labels.

7. Experiments

In this section we present the results of an extensive experimental validation of the proposed algorithm together with details about the considered benchmark.

7.1. Benchmark Datasets

To validate the performance of the IHMM-E and IHMM-B algorithms, we use two of the most important computer vision benchmarks: the Weizmann [32] and KTH [33] datasets for action recognition. For this problem, the class is the action depicted in the sequence (see for instance Figure 3). These data are footage material which requires a *feature extraction* procedure at the frame level. As shown in Figure 1, each frame is identified with a time step and the extracted features are the observable multivariate data. Our approach is based on histograms of oriented optical flows [34], a simple technique which describes the flows distribution in the whole frame as a histogram with 32 bins representing directions (Figure 4). Other benchmarks are also considered. CAD-60 [35] and SKIG [36] are, respectively, an activity and a gesture recognition dataset. The feature extraction algorithm proposed in [37] is used for both these datasets. The AUSLAN dataset [38] is based on gestures in the Australian sign language, while the JAPVOW dataset [39] contains speech data about Japanese vowels. Table 1 reports relevant information about these benchmark datasets. More specific information can be found on the cited references.

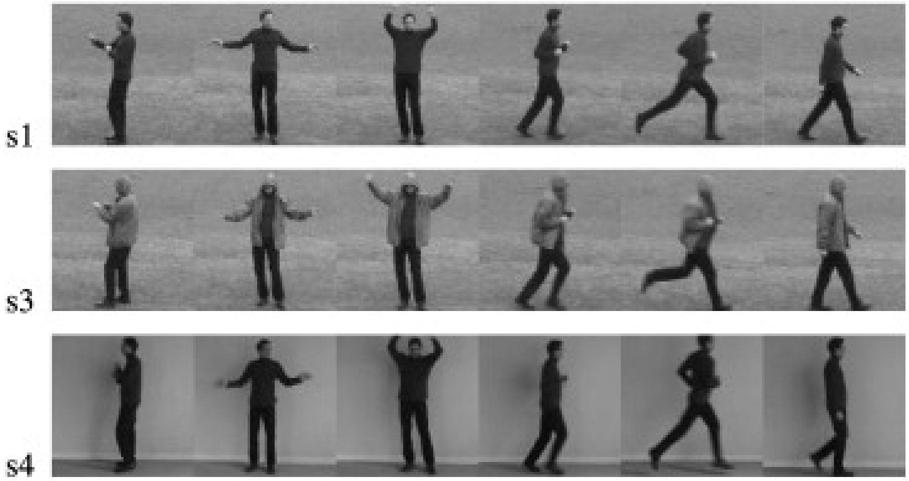

Figure 3: Frames extracted from the KTH dataset.

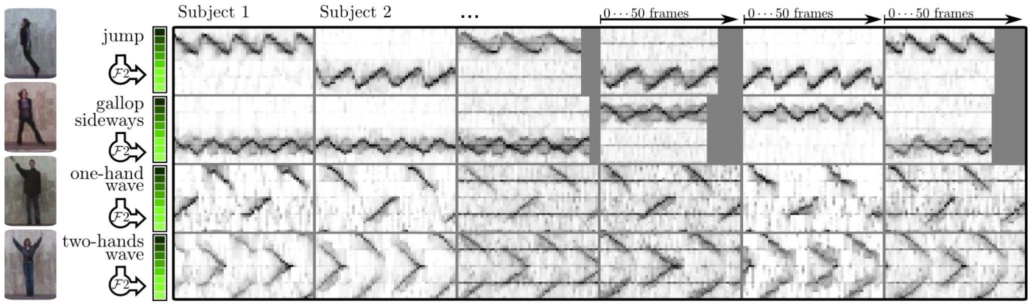

Figure 4: Low-level feature extraction. Rows correspond to different actions, columns to subjects. In each cell, feature values are shown as gray levels, with the different feature variables on the y axis, and frames on the x axis. Characteristic time-varying patterns are visible for each action.

To avoid features with small ranges being penalized by the *k*-NN classification with respect to others spanning larger domains, a feature normalization step is performed. This is a just a linear transformation in the feature space which makes the empirical mean of the sample equal to zero and the variance equal to one.

7.2. Results

The new IHMM-E and IHMM-B credal classifiers are empirically tested against the credal IHMM-L and the precise DTW classifiers on the ten datasets presented

Dataset	$\lvert C \rvert$	F	D	T
KTH_1	6	32	150	51
KTH_2	6	32	150	51
KTH_3	6	32	149	51
KTH_4	6	32	150	51
KTH	6	32	599	51
Weizmann	9	32	72	105-378
AUSLAN	95	22	1865/600	45-136
JAPVOW	9	12	370/270	7-29
SKIG	10	128	1080	100
CORNELL	11	96	60	350

Table 1: Datasets used for benchmarking. The columns denote, respectively, name, number of classes, number of features, size (test/training datasets sizes if no cross validation has been done) and the number of frames of each sequence (or their range if this number is not fixed). As usually done, the KTH dataset is also split in four subgroups.

in the previous section. A Matlab implementation of both IHMM-E and IHMM-B is used.[17] Regarding the multivariate version of DTW, the Matlab implementation described in [40] is used.

The real parameter of the imprecise Dirichlet model is fixed to $s = 1$, the number of hidden states to $M = 3$, and $k = 1$ is the value assumed for the k-NN algorithm. The choice of s is consistent with the suggestions in [3]. As already discussed in Section 3.2, the small value of M reflects the need of avoiding overly imprecise results. The choice of k has been already motivated at the end of Section 4.3. Moreover, as expected, tests with increasing values of k show a systematic degradation of the performance on our benchmark.

Regarding the separation between test and training set, we adopt the same configuration of the original publication of each dataset. Leave-one-out cross validation is considered for KTH, Weizmann and CORNELL, three-fold cross validation with ten runs for SKIG. A single run with fixed test and training set is considered instead for AUSLAN and JAPVOW.

Table 2 reports the determinacies and the normalized average output sizes of

[17] These tools are available as a free software at `http://ipg.idsia.ch/software`.

both the new algorithms and IHMM-L. These values are not measuring the performance but only the ability of these credal classifiers to return determinate results. As a comment, we see that the proposed methods (as well as IHMM-L) provide sufficiently informative outputs. An exception is IHMM-E on AUSLAN, JAPVOW and SKIG: the classifier is always indeterminate and the output includes (almost, in the case of JAPVOW and SKIG) all the classes. In these cases IHMM-E is therefore unable to discriminate over the different classes, and the resulting classification is not informative. Results for these special cases are therefore not significant and they are not reported in the following.

More generally, the higher determinacy of IHMM-B and IHMM-L when compared with IHMM-E should be related to the multivariate nature of the benchmark dataset: the higher is the dimensionality of the feature space the more likely are overlaps between the interval-valued distances between hyperboxes considered by IHMM-E, while the two other classifiers cope with one-dimensional descriptors, less prone to overlaps.

	IHMM-E		IHMM-B		IHMM-L	
	det	out	det	out	det	out
KTH_1	.24	.46	.85	.35	.70	.38
KTH_2	.60	.57	.47	.40	.56	.35
KTH_3	.11	.48	.75	.35	.82	.33
KTH_4	.50	.45	.83	.35	.60	.40
KTH	.10	.52	.58	.37	.60	.38
Weizmann	.26	.32	.68	.24	.77	.22
AUSLAN	.00	1.00	.67	.03	.93	.03
JAPVOW	.00	.97	.76	.28	.96	.23
SKIG	.00	.89	.79	.25	.00	.41
CORNELL	.25	.36	.93	.29	.75	.18

Table 2: Determinacies and normalized average output sizes for the benchmark datasets.

Information about the actual performance of the algorithms is in Tables 3 and 4. First consider the results in Table 3 about single and set accuracy. These refer to the accuracy of the classifiers when a single label is returned in the output and, if the classifier returns multiple labels, whether or not the right class belongs to this set. A fair comparison can be done only between the IHMM-B and IHMM-L, because of the similar values of determinacy and indeterminate output size. A clear outperformance by IHMM-B against IHMM-L is observed. Consider for instance the KTH dataset, the algorithms have almost the same determinacy, but the single accuracy is .780 for IHMM-B and only .299 for IHMM-L. Similarly, on the indeterminate instances, the output size is almost the same, but the set of classes includes the right class label in 87% of the cases for IHMM-B and 44.8% for IHMM-L. Similar comparisons cannot be done with IHMM-E. As an example, it is not obvious that a single accuracy equal to one for IHMM-E is better than the value .785 for IHMM-B, because of the higher determinacy of the second classifier. The interval-valued metrics discussed in Section 6.2 have been developed specifically for this kind of problems. Results of the performance according to these descriptors are reported in Table 4.

	IHMM-E		IHMM-B		IHMM-L	
	single	set	single	set	single	set
KTH_1	1.000	1.000	.812	.909	.301	.017
KTH_2	1.000	1.000	.600	.900	.180	.384
KTH_3	.941	.992	.750	.974	.070	.083
KTH_4	1.000	.993	.782	.923	.269	.524
KTH	1.000	1.000	.780	.870	.299	.448
Weizmann	1.000	1.000	.785	.876	.275	.143
AUSLAN	–	–	.852	.842	.021	.062
JAPVOW	–	–	.996	.989	.283	.462
SKIG	–	–	.928	.959	–	.450
CORNELL	.800	.840	.759	.500	.250	.071

Table 3: Single and set accuracies for the benchmark datasets.

As noted in Section 6.2, the interval $[u_{.65}, u_{.80}]$ provides a better summary of the performance of a credal classifier by also allowing for a fair comparison with a traditional (i.e., precise) classifier like DTW. The results in Table 4 show that the new methods clearly outperform IHMM-L. Moreover, IHMM-B is always better or equal to IHMM-E. Impressively, IHMM-B also competes with DTW: the average rank is 1.3 for IHMM-B and 2.5 for DTW. This shows the quality of our approach and the (known) degradation of the DTW performance in the multiple-features case.

On the basis of the above experiments, we regard IHMM-B as the algorithm of choice for credal classification of multivariate time series.

	IHMM-E		IHMM-B		IHMM-L		DTW
	$u_{.65}$	$u_{.80}$	$u_{.65}$	$u_{.80}$	$u_{.65}$	$u_{.80}$	acc
KTH_1	.636	.739	**.778**	**.797**	.211	.212	.613
KTH_2	**.480**	**.598**	**.553**	**.621**	.201	.225	.369
KTH_3	.556	.675	**.716**	**.753**	.073	.076	.529
KTH_4	.544	.675	**.745**	**.769**	.281	.310	.480
KTH	.535	.653	**.671**	**.725**	.283	.309	.525
Weizmann	**.626**	**.725**	**.706**	**.747**	.236	.242	.540
AUSLAN	–	–	.725	.763	.021	.022	**.838**
JAPVOW	–	–	**.889**	**.923**	.283	.285	.697
SKIG	–	–	.850	.880	.155	.204	**.957**
CORNELL	.472	.544	**.722**	**.725**	.197	.200	.283

Table 4: Accuracies for the benchmark datasets. Best performances are boldfaced.

IHMM-B has a precise counterpart obtained by setting $s = 0$ in the constraints of the imprecise Dirichlet model.[18] We evaluate the accuracy of this precise classifier on the whole test set. Then we compute the accuracy only on the instances for which IHMM-B is indeterminate. We also consider the accuracy of the precise classifier on the instances for which IHMM-B is determinate. This is equal to the single accuracy of IHMM-B as in Table 3. In fact, by definition, when a credal classifier returns a single class, this coincides with the one returned by its precise counterpart. Figure 5 depicts a comparison, for all the benchmark datasets, of the accuracy of this precise counterpart on the whole dataset (gray histograms), on the indeterminate instances (black histograms), and on the determinate ones (white histograms).

[18] A precise counterpart of a credal classifier based on the imprecise Dirichlet model can be also obtained by taking any single prior consistent with it.

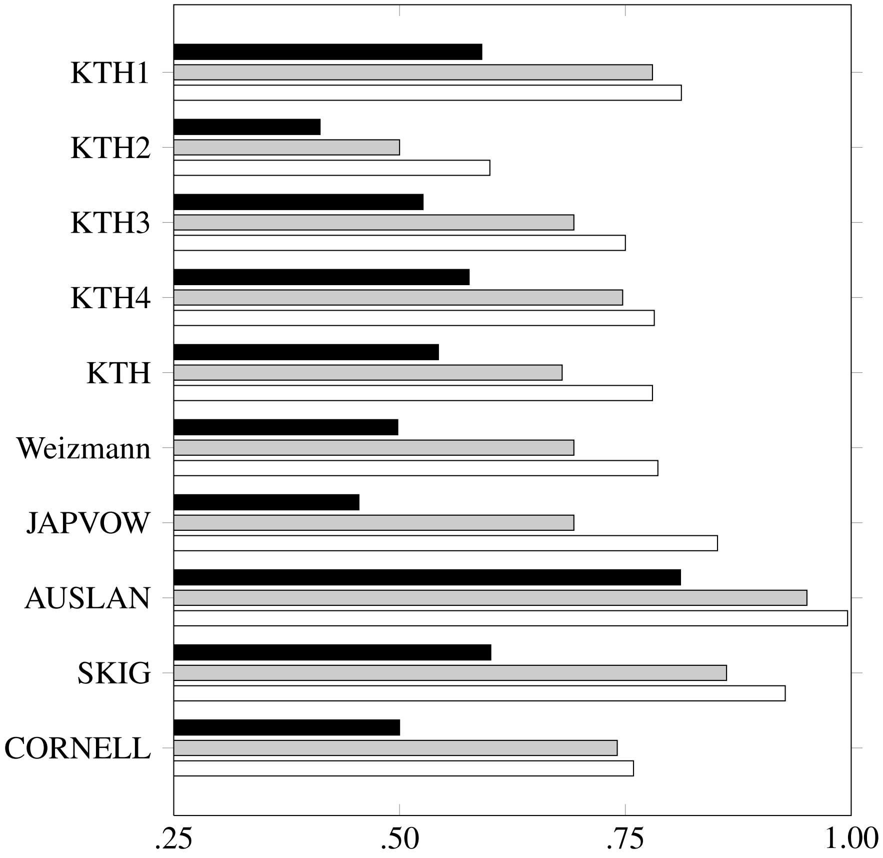

Figure 5: Accuracies of the precise counterpart of IHMM-B on: (i) all the test instances (gray); (ii) the instances for which IHMM-B is indeterminate (black); (iii) the instances for which IHMM-B is determinate (white).

As expected (see the discussion in Section 1), the accuracy on the whole set of instances is always higher than the corresponding value for the indeterminate instances only, and smaller for the determinate ones.

IHMM-B is therefore effective in discriminating hard-to-classify from "easy" instances. In other words, if this credal classifier returns a single class label, we can reasonably expect that this is the correct one, while if multiple outputs are reported we should definitely prefer this indeterminacy to the single output returned by a precise classifier, which is more likely to be wrong.

8. Conclusions and Outlooks

Two novel credal classifiers for multivariate temporal data have been presented. Imprecise HMMs are learned from each sequence. The first classifier summarizes the model with a hyperbox in the feature space. This datum is classified by a generalization of the k-NN approach. The second classifier uses an interval-valued dissimilarity measure. The second approach has the better performance: it outperforms a credal classifier previously proposed for this task and compete with the state-of-the-art methods. In future work, we want to investigate novel, more reliable, learning techniques such as the likelihood-based approach already considered for complete data in [41]. Also alternative approaches to the evaluation of the dissimilarity level between imprecise HMMs (e.g., the KL divergence) should be considered.

Appendix A. Linearly Constrained Quadratic Optimization

Let us consider the linearly constrained quadratic optimization tasks to be solved by the IHMM-B algorithm. These task can be solved in polynomial (roughly cubic) time because the solution lies on an extreme point of the feasible region [42].

The minimization of the objective function in Equation (24) rewrites as:

$$\min_{\tilde{P}(X)\in\tilde{K}(X),\tilde{Q}(X')\in\tilde{K}(X')} \sum_{x\in\mathcal{X},x'\in\mathcal{X}'} k(x,x')\cdot\tilde{P}(x)\cdot\tilde{Q}(x'). \tag{A.1}$$

We can easily prove that the solution of this problem, denoted as $[\tilde{P}^*(X),\tilde{Q}^*(X')]$, corresponds to an extreme point of the feasible region, i.e., $\tilde{P}^*(X)\in\mathrm{ext}[\tilde{K}(X)]$ and $\tilde{Q}^*(X')\in\mathrm{ext}[\tilde{K}(X')]$.

To do that, let us add the additional constraint $\tilde{Q}(X') = \tilde{Q}^*(X')$. This makes the problem a linear program as the objective function becomes:

$$\sum_{x\in\mathcal{X}} \tilde{P}(x)\cdot\left[\sum_{x'\in\mathcal{X}'} \tilde{Q}^*(x')\cdot k(x,x')\right], \tag{A.2}$$

with the linear constraints $\tilde{P}(X) \in \tilde{K}(X)$. The solution of this linear program coincides with the optimal solution $\tilde{P}^*(X)$. On the other side, the solution should also be an extreme point of the feasible region. Thus, $\tilde{P}^*(X) \in \mathrm{ext}[\tilde{K}(X)]$. We can similarly prove that $\tilde{Q}^*(X') \in \mathrm{ext}[\tilde{K}(X')]$. A similar result holds even if we consider the maximum instead of the minimum.

Appendix B. Computation of the Stationary Credal Set

Given an imprecise Markov chain as in Section 2, for each $\mathcal{X}' \subseteq \mathcal{X}$, define $Q_{\mathcal{X}'} : \mathcal{X} \to \mathbb{R}$, such that, $\forall x \in \mathcal{X}$:

$$\overline{Q}_{\mathcal{X}'}(x) := \min\left\{\sum_{x'\in\mathcal{X}'} \overline{P}(x'|x), 1 - \sum_{x'\in\mathcal{X}\setminus\mathcal{X}'} \underline{P}(x'|x)\right\}. \tag{B.1}$$

Given this function, $\forall g : \mathcal{X} \to \mathbb{R}$, define $\overline{R}_g : \mathcal{X} \to \mathbb{R}$, such that:

$$\overline{R}_g(x) := \underline{g} + \int_{\underline{g}}^{\overline{g}} \overline{Q}_{\{x'\in\mathcal{X}:g(x')\geq t\}}(x)\mathrm{d}t, \tag{B.2}$$

for each $x \in \mathcal{X}$, with $\underline{g} := \min_{x\in\mathcal{X}} g(x)$ and $\overline{g} := \max_{x\in\mathcal{X}} g(x)$. Proceed similarly for the unconditional probability of the first hidden variable. In this way the following numbers (instead of functions) are defined:

$$\overline{Q}^0_{\mathcal{X}'} := \min\left\{\sum_{x\in\mathcal{X}'} \overline{P}(x'), 1 - \sum_{x\in\mathcal{X}'} \underline{P}(x')\right\}. \tag{B.3}$$

$$\overline{R}^0_g := \underline{g} + \int_{\underline{g}}^{\overline{g}} \overline{Q}^0_{\{x'\in\mathcal{X}:g(x')\geq t\}}\mathrm{d}t. \tag{B.4}$$

A "lower" version of these functions and numbers can be obtained by simply replacing the lower probabilities with the uppers, maxima with the minima, and vice versa. For each $i = 1, \dots, n$ let $h_i : \mathcal{X} \to \mathbb{R}$. To characterize the stationary credal set $\tilde{K}(X)$, consider $\overline{P}^*(x') := \max_{P(X) \in \tilde{K}(X)} P(x')$. Given the recursion:

$$h_{j+1}(x) := \overline{R}_{h_j}(x), \tag{B.5}$$

with initialization $h_1 := I_{x'}$[19], we obtain:

$$\overline{P}^*(x') := \lim_{n \to \infty} \overline{R}^0_{h_n}, \tag{B.6}$$

and similarly for the upper.

Acknowledgements

We thank Marco Zaffalon for suggesting us the idea of using expected counts in the imprecise Dirichlet model to learn credal sets from incomplete data.

References

[1] A. Antonucci, R. de Rosa, A. Giusti, F. Cuzzolin, Temporal data classification by imprecise dynamical models, in: F. Cozman, T. Denoeux, S. Destercke, T. Seidenfeld (Eds.), Proceedings of the Eighth International Symposium on Imprecise Probability: Theories and Applications (ISIPTA '13), SIPTA, 2013, pp. 13–22.

[2] P. Walley, Statistical Reasoning with Imprecise Probabilities, Chapman and Hall, 1991.

[3] P. Walley, Inferences from multinomial data: Learning about a bag of marbles, Journal of the Royal Statistical Society, Series B 58 (1) (1996) 3–34.

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

[5] M. Zaffalon, E. Fagiuoli, Tree-based credal networks for classification, Reliable Computing 9 (6) (2003) 487–509.

[19] For each $x' \in \mathcal{X}$, $I_{x'}$ is the *indicator function* of x', i.e., a function $\mathcal{X} \to \mathbb{R}$ such that $I_{x'}(x)$ is equal to one if $x = x'$ and zero otherwise.

[6] G. Corani, A. Antonucci, Credal ensembles of classifiers, Computational Statistics and Data Analysis 71 (2014) 818–831.

[7] A. Antonucci, M. Cattaneo, G. Corani, Likelihood-based robust classification with Bayesian networks, in: Communications in Computer and Information Science, Vol. 299(5) of Advances in Computational Intelligence, Springer, Berlin / Heidelberg, 2012, pp. 491–500.

[8] A. Antonucci, R. de Rosa, A. Giusti, Action recognition by imprecise hidden Markov models, in: Proceedings of the 2011 International Conference on Image Processing, Computer Vision and Pattern Recognition (IPCV 2011), CSREA Press, 2011, pp. 474–478.

[9] G. de Cooman, F. Hermans, A. Antonucci, M. Zaffalon, Epistemic irrelevance in credal networks: the case of imprecise Markov trees, International Journal of Approximate Reasoning 51 (9) (2010) 1029–1052.

[10] D. Mauá, C. de Campos, A. Antonucci, Algorithms for hidden Markov models with imprecisely specified parameters, in: Proceedings of the Brazilian Conference on Intelligent Systems (BRACIS 2014), 2014, accepted for publication.

[11] P. Smyth, Clustering sequences with hidden Markov models, in: Advances in Neural Information Processing Systems, MIT Press, 1997, pp. 648–654.

[12] A. Van Camp, G. de Cooman, A new method for learning imprecise hidden Markov model, in: S. Greco, B. Bouchon-Meunier, G. Coletti, B. Matarazzo, R. Yager (Eds.), Communications in Computer and Information Science, Vol. 299, Springer, 2012, pp. 460–469.

[13] G. de Cooman, F. Hermans, E. Quaeghebeur, Sensitivity analysis for finite Markov chains in discrete time, in: Proceedings of the Twenty-Fourth Conference on Uncertainty in Artificial Intelligence (UAI 2008), 2008, pp. 129–136.

[14] D. Škulj, Discrete time Markov chains with interval probabilities, International Journal of Approximate Reasoning 50 (8) (2009) 1314–1329.

[15] M. Zaffalon, G. Corani, D. Mauá, Utility-based accuracy measures to empirically evaluate credal classifiers, in: F. Coolen, G. de Cooman, T. Fetz,

M. Oberguggenberger (Eds.), Proceedings of the Seventh International Symposium on Imprecise Probability: Theories and Applications (ISIPTA '11), SIPTA, Innsbruck, 2011, pp. 401–410.

[16] G. A. Ten Holt, M. J. T. Reinders, E. Hendriks, Multi-dimensional dynamic time warping for gesture recognition, in: G. Zaverucha, A. Costa (Eds.), Proceedings of the 13th annual conference of the Advanced School for Computing and Imaging, 2007, pp. 23–32.

[17] D. Koller, N. Friedman, Probabilistic Graphical Models: Principles and Techniques, MIT Press, 2009.

[18] I. Levi, The Enterprise of Knowledge. An Essay on Knowledge, Credal Probability, and Chance, MIT Press, Cambridge, 1980.

[19] A. Antonucci, M. Zaffalon, Decision-theoretic specification of credal networks: a unified language for uncertain modeling with sets of Bayesian networks, International Journal of Approximate Reasoning 49 (2) (2008) 345–361.

[20] D. Mauá, C. de Campos, A. Benavoli, A. Antonucci, On the complexity of strong and epistemic credal networks, in: Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence, 2013, pp. 391–400.

[21] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society, Series B 39 (1) (1977) 1–38.

[22] G. de Cooman, M. Zaffalon, Updating beliefs with incomplete observations, Artificial Intelligence 159 (1–2) (2004) 75–125.

[23] T. Shi, M. Belkin, B. Yu, Data spectroscopy: Eigenspaces of convolution operators and clustering, The Annals of Statistics 37 (6B) (2009) 3960–3984.

[24] R. Crossman, D. Škulj, Imprecise Markov chains with absorption, International Journal of Approximate Reasoning 51 (9) (2010) 1085–1099.

[25] A. Antonucci, F. Cuzzolin, Credal sets approximation by lower probabilities: Application to credal networks, in: E. Hüllermeier, R. Kruse, F. Hoffmann (Eds.), Computational Intelligence for Knowledge-Based Systems Design, 13th International Conference on Information Processing and Management

of Uncertainty (IPMU 2010), Springer, Dortmund, Germany, 2010, pp. 716–725.

[26] A. Antonucci, An interval-valued dissimilarity measure for belief functions based on credal semantics, in: T. Denoeux, M. Masson (Eds.), Belief Functions: Theory and Applications - Proceedings of the second International Conference on Belief Functions, Vol. 164 of Advances in Soft Computing, Springer, 2012, pp. 37–44.

[27] A. Antonucci, A. Karlsson, D. Sundgren, Decision making with hiearchical credal sets, in: Proceedings of the 15th Information Processing and Management of Uncertainty in Knowledge-Based Systems Conference (IPMU 2014), 2014, accepted for publication.

[28] E. Keogh, C. A. Ratanamahatana, Exact indexing of dynamic time warping, Knowledge and Information Systems 7 (3) (2005) 358–386.

[29] J. Hershey, P. Olsen, Variational Bhattacharyya divergence for hidden Markov models, in: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2008), 2008, pp. 4557–4560.

[30] G. Coleman, H. Andrews, Image segmentation by clustering, Proceedings of the IEEE 67 (5) (1979) 773–785.

[31] L. Utkin, F. Coolen, Interval-valued regression and classification models in the framework of machine learning, in: Proceedings of the Seventh International Symposium on Imprecise Probability: Theories and Applications (ISIPTA '11), SIPTA, 2011, pp. 371–380.

[32] L. Gorelick, M. Blank, E. Shechtman, M. Irani, R. Basri, Actions as space-time shapes, Transactions on Pattern Analysis and Machine Intelligence 29 (12) (2007) 2247–2253.

[33] C. Schuldt, I. Laptev, B. Caputo, Recognizing human actions: a local SVM approach, in: Proceedings of 17th International Conference on Pattern Recognition (ICPR 2004), 2004.

[34] R. Chaudhry, A. Ravichandran, G. Hager, R. Vidal, Histograms of oriented optical flow and Binet-Cauchy kernels on nonlinear dynamical systems for the recognition of human actions, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009), 2009.

[35] J. Sung, C. Ponce, B. Selman, A. Saxena, Unstructured human activity detection from rgbd images, in: Proceedings of the International Conference on Robotics and Automation (ICRA 2012), 2012.

[36] L. Liu, L. Shao, Learning discriminative representations from rgb-d video data, in: Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence (IJCAI'13), AAAI Press, 2013, pp. 1493–1500.

[37] S. Fanello, I. Gori, G. Metta, F. Odone, Keep it simple and sparse: Real-time action recognition, Journal of Machine Learning Research 14 (1) (2013) 2617–2640.

[38] J. Kies, Empirical methods for evaluating video-mediated collaborative work, Ph.D. thesis, Virginia Tech (March 1997).

[39] J. T. M. Kudo, M. Shimbo, Multidimensional curve classification using passing-through regions, Pattern Recognition Letters 20 (1999) 1103–1111.

[40] P. Sanguansat, Multiple multidimensional sequence alignment using generalized dynamic time warping, WSEAS Transactions on Mathematics 11 (2012) 668–678.

[41] A. Antonucci, M. Cattaneo, G. Corani, Likelihood-based naive credal classifier, in: F. Coolen, G. de Cooman, T. Fetz, M. Oberguggenberger (Eds.), Proceedings of the Seventh International Symposium on Imprecise Probability: Theories and Applications (ISIPTA '11), SIPTA, Innsbruck, 2011, pp. 21–30.

[42] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, New York, NY, USA, 2004.