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 meaning that multiple class labels can be returned in output. Experiments on benchmark datasets for computer vision show that these methods achieve the required robustness whilst outperforming other precise and imprecise methods.

Keywords: 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 (Antonucci et al., 2013).

[*] 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 - Galleria 2 - Via Cantonale - Manno (Switzerland).

[2] Dipartimento di Scienze dellInformazione, 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* (Walley, 1991) extends Bayesian theory of subjective probability to cope with sets of distributions, this potentially providing 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* (Walley, 1996). An example, namely the first attempt in this direction, is the *naive credal classifier* (Zaffalon, 2002), this generalizing the naive Bayes classifier to imprecise probabilities. While the imprecise Dirichlet model copes with multiple priors, precise classifiers quantified with different priors might assign different class labels to the same instance: in these cases the imprecise classifier returns multiple 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 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 can be useful preprocessing tools, assigning the right class label to prior-independent instances and partially suspending the judgement (to be demanded to other tools) otherwise.

Despite the relatively high number of credal classifiers proposed in the literature, no models of this kind, 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 (de Cooman et al., 2008, 2010). As a matter of fact, Markov models in

[4] Besides the naive, other examples of credal classifiers proposed in the literature include models with more complex topologies (Zaffalon and Fagiuoli, 2003) and imprecise averages of precise models (Corani and Antonucci, 2012). Alternative quantification techniques not based on the imprecise Dirichlet model have been proposed for general topologies by Antonucci et al. (2012).

[5] The only exception is the previous work of the authors in (Antonucci et al., 2011b). The algorithms in the present paper are a natural evolution of those approaches (and numerical tests showing much better performances of the new methods are reported in Section 7).

their precise formulation have been often applied to classification of time series (e.g., Smyth, 1997), 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 (e.g., because referred to hidden variables) data and, more marginally, by the lack of suitable inference algorithms.

It therefore looks almost 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 (Antonucci et al., 2011b; Van Camp and de Cooman, 2012) 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 are known to converge to a condition of *stationarity*, and this holds even in the imprecise case (de Cooman et al., 2008). 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 indeed an *imprecise mixture* (i.e., with multiple specification of the weights) of Gaussian densities over the observable variable. Two novel algorithms are proposed to perform classification with these models. The first, called IHMM-KNN, evaluates the mixture expected value, which is used as a static attribute for a standard classification problem. The second, called IHMM-BHATT, assumes 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 w.r.t. the imprecise parameters are estimated. This is done efficiently by solving a linear (for IHMM-KNN) and a quadratic (for IHMM-BHATT) optimization task.

After this step, IHMM-KNN 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 interval data is developed. The same approach can be used to process the interval-valued distances among sequences returned by IHMM-BHATT. Both the algorithms are credal classifiers for time series, possibly assigning more than a class label to a sequence. Performances are tested on some of the most important computer vision benchmarks. The results are promising: 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 (Zaffalon et al., 2011) to compare performances of credal and traditional classifiers.

The performance is also good when compared with the *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 (Ten Holt et al., 2007), 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 first 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-KNN and IHMM-BHATT 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 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 time-homogeneous, 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* and it is known to assume a stationary behaviour on long sequences, i.e., the following limit exists:

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

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 (3), 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.[6] 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*:

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

[6] 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.

which evaluates the overlap between statistical samples generated by the two distributions. In the remaining of this paper we will develop two algorithms based, respectively, on the descriptor in Equation (5) and on the distance in Equation (6). These ideas will be extended to the imprecise-probabilistic framework, taking into account the case of multivariate observations. Overall, this will lead 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 Levi (1980), 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 meaning 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 $x_{t-1} \in \mathcal{X}$.[7] 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 $T \to \infty$. The limit behaviour of imprecise Markov chains has been studied in de Cooman et al. (2008), where a formula to compute $\tilde{K}(X)$ has been derived.[8] This formula is reported in Appendix B.

[7] This joint credal set is also called the *strong extension* of the imprecise Markov chain. As shown by Proposition 1 in Antonucci and Zaffalon (2008), this credal set can be obtained by considering only the extreme points of the credal sets and then taking the convex hull.

[8] 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 Mauá et al. (2013).

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

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

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.[9] 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 Dempster et al. (1977) is the 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 missing values 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. Yet, the imprecise Dirichlet model, commonly used to learn credal sets, needs the data to be complete, and no alternatives for incomplete data are available.[10]

[9] 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 on the hidden layer are in Section 3.2.

[10] The conservative updating rule proposed by de Cooman and Zaffalon (2004) 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 too imprecise results.

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. 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 compute, for each $t = 2, \ldots, T$, the probability that $X_t = x$ and multiply 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 very 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 $\mathbf{o}_1, \ldots, \mathbf{o}_T$, with $\mathbf{o}_t \in \mathbb{R}^F$ 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:

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

where o_t^f is the f-th component of the array $\mathbf{o}_t$, $\mathcal{N}_\sigma^\mu$ 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 $X_t = x_t$.[11]

The clustering-based technique proposed by Shi et al. (2009) defines a possible initialization of for the emission terms in the EM, while uniform choices are adopted for the transition and the prior. Overall, 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 on 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. IHMM-KNN: Credal Classification of Time Series

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 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 by de Cooman et al. (2008), its computation, which is briefly summarized in Appendix B, can be obtained by a Choquet integration. Thus, in this generalized setup, distribution $\tilde{P}(X)$ in Equation (5) is only required to belong to $\tilde{K}(X)$. Note that $\tilde{K}(X)$ is a finitely generated credal set which can be equivalently characterized by (a finite number

[11] 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.

of) linear constraints. Regarding the emission terms, nothing changes as they are assumed to be precise. Thus, for each feature o_f, with $f = 1, \dots, F$, 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 $\underline{\mathbf{o}}, \overline{\mathbf{o}} \in \mathbb{R}^F$ are indeed obtained by applying the optimization is 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.

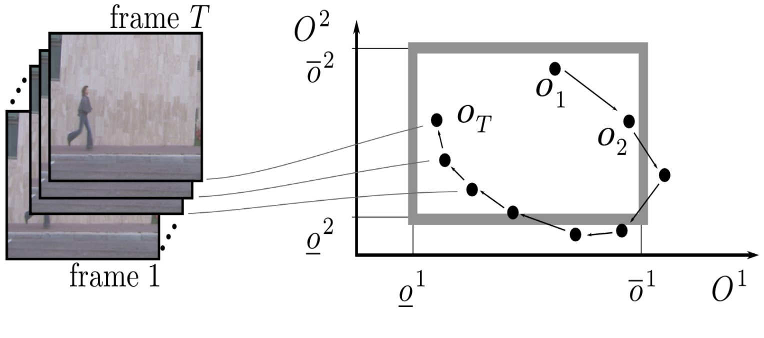

Figure 1: From trajectories to hyperboxes in the feature space. The example refers to footage data from which two features are extracted at the frame level.

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 $\mathbf{x}, \mathbf{y} \in \mathbb{R}^F$, defines their distance δ as

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

Given two points $\underline{\mathbf{x}}, \overline{\mathbf{x}} \in \mathbb{R}^F$ 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 $[\underline{\mathbf{x}}, \overline{\mathbf{x}}]$ and defined as

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

The problem of extending a distance defined over points to hyperboxes can be solved by considering the general ideas proposed in Antonucci (2012).

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 boxes is:

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

and similarly, with the maximum instead of the minimum for the upper distance $\overline{\delta}([\underline{\mathbf{x}}, \overline{\mathbf{x}}], [\underline{\mathbf{y}}, \overline{\mathbf{y}}])$. 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:

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

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 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.[12] 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 extend 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 $\{c^d, \mathbf{x}^d\}_{d=1}^D$ classification is intended as the problem of assigning a class label $\tilde{c} \in \mathcal{C}$ to a new instance $\tilde{\mathbf{x}}$ on the basis of the data. The k-NN algorithm for $k = 1$ assigns to $\tilde{\mathbf{x}}$ the label associated with the instance nearest to $\tilde{\mathbf{x}}$, i.e., the solution is $\tilde{c} := c^{d^*}$ with

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

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. Yet, 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

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

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 controversial. 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 class in output. If the set of optimal classes according to this criterion is defined as $\mathcal{C}^*$, we have that $c \in \mathcal{C}^*$ if and only if there exists a datum $(c^i, \mathbf{x}^i)$ such that $c = c^i$ and

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

for each $d = 1, \ldots, D$ such that $c^d \neq c^i$. 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. Note also that the case $k > 1$ simply requires the iteration of the evaluation in Equation (19).

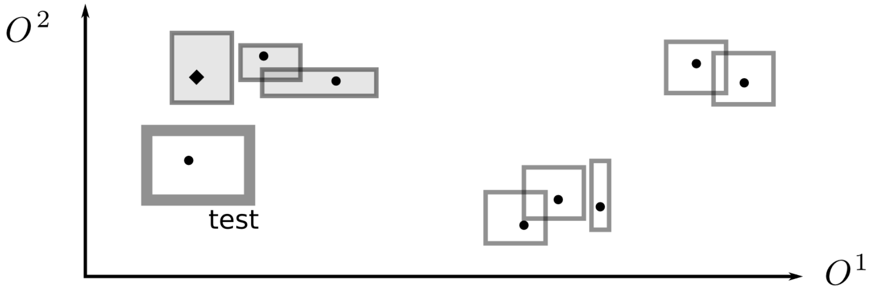

Figure 2: Rectangular data processed by the 1-NN classifier. Gray background denotes data whose interval distance from the test instance is undominated. Points inside the rectangles describe consistent precise data and the diamond is the nearest instance.

4.4. The IHMM-KNN Credal Classifier

By merging the discussions in Sections 3, 4.1, 4.2 and 4.3 we have a classifier for time series, to be called IHMM-KNN, based on imprecise HMMs. In summary, for 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 kNN credal classification (Section 4.3) based on the interval-valued distance between hyperboxes (Section 4.2).

5. Directly Coping with the Distributions: IHMM-BHATT

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), while classification is not suffering 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 to 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 $\tilde{P}(\mathbf{O})$ such that:

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

for each $\mathbf{o} \in \mathbb{R}^F$. 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 intended in $\mathbb{R}^F$). In general, such a distance cannot be computed analytically, but a good approximation can be easily obtained as suggested by Hershey and Olsen (2008). 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:

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

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:

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

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

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

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. The class label assigned to the test instance is the one of the training instance at minimum distance exactly as in Equation (18).

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 very 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 optimize 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 $\tilde{Q}(X) \in \tilde{K}(X')$. 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.[13] We call this credal classifier IHMM-BHATT.

[13] Numerical problems can be encountered because of too small coefficient in the objective function in Equation (24). 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.

6. Related Works, Complexity, and Performance Evaluation

Another credal classifier for time series based on imprecise HMMs, and called here IHMM-LIK, has been proposed in Antonucci et al. (2011b). 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 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 (i.e., $F > 1$) case Ten Holt et al. (2007). Both these methods will be compared with our classifiers 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 Utkin and Coolen (2011) can be used to define a 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 dimensionality. This is not the case for IHMM-KNN and IHMM-BHATT, whose complexity is analyzed below.

6.1. Complexity Analysis

Let us first evaluate IHMM-KNN. Our approach to the learning of imprecise HMMs has the same time complexity of 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-BHATT, 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 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* (*det*), i.e., percentage of instances classified with a single label; the *average output size* (*out*), i.e., average number of classes on instances for which multiple labels are returned. For accuracy we distinguish between: *single-accuracy* (*sing-acc*), i.e., accuracy over instances classified as a single label; *set-accuracy* (*set-acc*), i.e., the accuracy over the instances classified with multiple labels[14].

A utility-based measure has been recently proposed in Zaffalon et al. (2011) 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 (Zaffalon et al., 2011). 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 rank performances.

The precise counterpart of a credal classifier is a classifier always returning a single class included in the output of the credal classifier. As an example, both IHMM-KNN and IHMM-BHATT 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* (*p-sing-acc*), i.e., the accuracy of the precise classifier when the credal returns a single label; the *precise set-accuracy* (*p-set-acc*), i.e., the accuracy of the precise classifier when the credal returns multiple labels.

[14] In this case, classification is considered correct if the set of labels includes the true class.

7. Experiments

7.1. Benchmark Datasets

To validate the performance of the IHMM-KNN and IHMM-BHATT algorithms, we use two of the most important computer vision benchmarks: the Weizmann (Gorelick et al., 2007) and KTH (Schuldt et al., 2004) datasets for *action recognition*. For this problem, the class is the action depicted in the sequence (see for instance Figure 3).

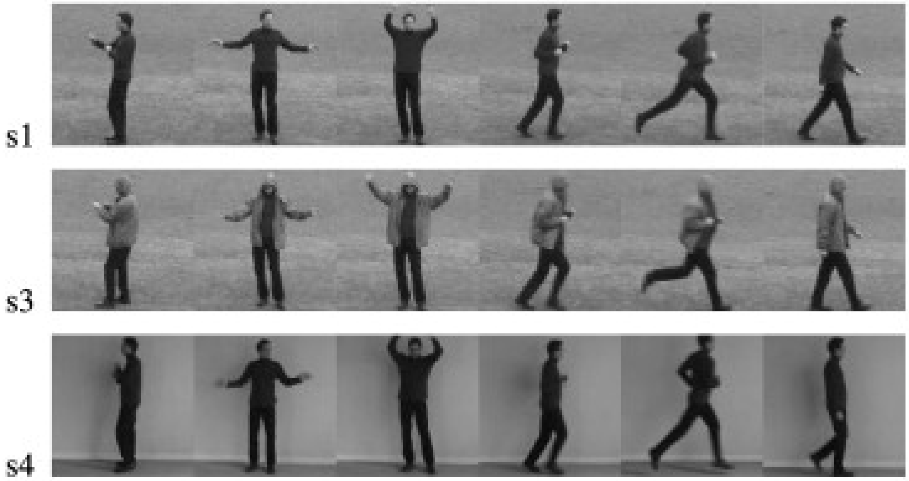

Figure 3: Frames extracted from the KTH dataset.

These data are footage material which requires a *features extraction* procedure at the frame level. Our approach is based on histograms of oriented optical flows Chaudhry et al. (2009), a simple technique which describes the flows distribution in the whole frame as an histogram with 32 bins representing directions (Figure 4).

For a through validation also the AUSLAN dataset (Kies, 1997) based on gestures in the Australian sign language and the JAPVOW dataset (M. Kudo and Shimbo, 1999) with speech about Japanese vowels are considered. Table 1 reports relevant information about these benchmark datasets.

To avoid features with small ranges being penalized by the k-NN with respect to others spanning larger domains a feature normalization step has been 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.

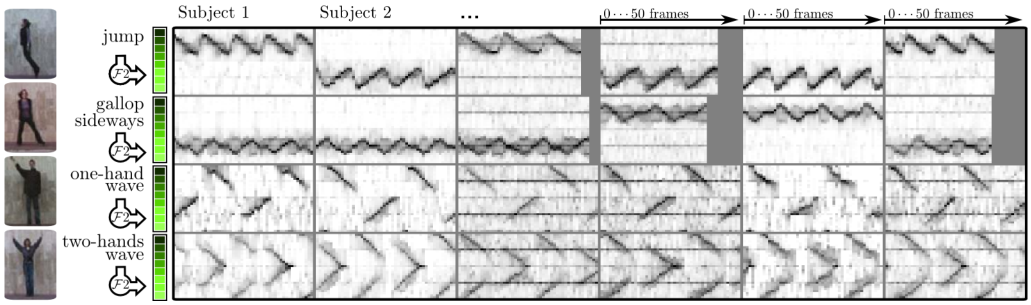

Figure 4: Low-level feature extraction. Rows correspond to different actions (i.e., class labels), columns to subjects. In each cell, feature values are shown as gray levels, with the different feature variables on the y axis, and frames on the x axis. Characteristic time-varying patterns are visible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

Dataset	$\lvert\mathcal{C}\rvert$	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

Table 1: Datasets used for benchmarking. The columns denotes, 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.

7.2. Results

The new IHMM-KNN and IHMM-BHATT credal classifier are empirically tested against the credal IHMM-LIK and the precise DTW classifier on the seven datasets described in the previous section. Five runs of ten-fold cross validation are considered for KTH and Weizmann. A single run with fixed test and training set is considered instead for AUSLAN and JAPVOW. We implemented in Matlab both IHMM-kNN and IHMM-BHATT.[15] Regarding DTW, the Mathworks implementation for Matlab has been used.

The real parameter s of the imprecise Dirichlet model has been fixed to $s = 1$, the number of hidden states M has been fixed to $M = 3$, and $k = 1$ is the value assumed for the k-NN algorithm.

Table 2 reports determinacies and average output sizes of both our algorithms and IHMM-LIK. These values are not measuring the performance of the classifiers but only their ability to return more or less determinate outputs. As a comment, we see that the proposed methods (as well as IHMM-LIK) are mostly successful in providing informative outputs. The exception is IHMM-KNN on AUSLAN and JAPVOW: the IHMM-KNN classifier is always indeterminate and the output includes (almost, in the case of JAPVOW) all the classes. In these case IHMM-KNN is therefore unable to discriminate over the different classes, and the resulting classifier is not informative. Results for these special case are therefore not significant and they will not be reported in the following.

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

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 output (sing-acc) 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-BHATT and IHMM-LIK, because of the similar values of determinacy and indeterminate output size. A clear outperformance of IHMM-BHATT against IHMM-LIK is observed. Consider for instance the KTH dataset, the algorithms have almost the

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

Dataset	IHMM-KNN		IHMM-BHATT		IHMM-LIK	
	det	*out*	*det*	*out*	*det*	*out*
KTH$_1$	24%	2.8	85%	2.1	70%	2.3
KTH$_2$	6%	3.4	47%	2.4	56%	2.1
KTH$_3$	11%	2.9	75%	2.1	82%	2.0
KTH$_4$	5%	2.7	83%	2.1	60%	2.4
KTH	10%	3.1	58%	2.2	60%	2.3
Weizmann	26%	2.9	68%	2.2	77%	2.0
AUSLAN	0%	95	67%	2.7	93%	2.4
JAPVOW	0%	8.7	76%	2.5	96%	2.1

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

same determinacy, but the single-accuracy is 78% for IHMM-BHATT and only 29.9% for IHMM-LIK. 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-BHATT and 44.8% for IHMM-LIK. Similar comparisons cannot be done with IHMM-KNN. As an example, it is not obvious that a 100% single-accuracy of IHMM-KNN is better than the 78.5% of IHMM-BHATT, because of the higher determinacy of the second classifier. The interval-valued metrics discussed in Section 6.2 have been developed precisely to cope with this problem. Results of the performance according to this indicator are reported in Table 4.

Dataset	IHMM-KNN		IHMM-BHATT		IHMM-LIK	
	sing-acc	*set-acc*	*sing-acc*	*set-acc*	*sing-acc*	*set-acc*
KTH$_1$	100%	100%	81.2%	90.9%	30.1%	1.7%
KTH$_2$	100%	100%	60.0%	90.0%	18.0%	38.4%
KTH$_3$	94.1%	99.2%	75.0%	97.4%	7.0%	8.3%
KTH$_4$	100%	99.3%	78.2%	92.3%	26.9%	52.4%
KTH	100%	100%	78.0%	87.0%	29.9%	44.8%
Weizmann	100%	100%	78.5%	87.6%	27.5%	14.3%
AUSLAN	–	–	85.2%	84.2%	2.1%	6.2%
JAPVOW	–	–	99.6%	98.9%	28.3%	46.2%

Table 3: Single and set accuracies on the benchmark.

As noted in Section 6.2, the interval $[u_{.65}, u_{.80}]$ provides a better summary of the credal classifiers performance by also allowing for a comparison with a traditional classifier like DTW. The results in Table 4 show that the new methods clearly outperform IHMM-LIK. Moreover, IHMM-BHATT is always better or equal to IHMM-KNN and should be therefore regarded as the algorithm of choice for these problems. Impressively, our methods also compete with the DTW, showing both the quality of our approach and the (known) degradation of the DTW performance in the multiple-features case.

Dataset	IHMM-KNN		IHMM-BHATT		IHMM-LIK		DTW
	$u_{.65}$	$u_{.80}$	$u_{.65}$	$u_{.80}$	$u_{.65}$	$u_{.80}$	*acc*
KTH$_1$	63.6%	73.9%	**77.8%**	**79.7%**	21.1%	21.2%	61.3%
KTH$_2$	**48.0%**	**59.8%**	**55.3%**	**62.1%**	20.1%	22.5%	36.9%
KTH$_3$	55.6%	67.5%	**71.6%**	**75.3%**	7.3%	7.6%	52.9%
KTH$_4$	54.4%	67.5%	**74.5%**	**76.9%**	28.1%	31.0%	48.0%
KTH	53.5%	65.3%	**67.1%**	**72.5%**	28.3%	30.9%	52.5%
Weizmann	**62.6%**	**72.5%**	**70.6%**	**74.7%**	23.6%	24.2%	54.0%
AUSLAN	–	–	72.5%	76.3%	2.1%	2.2%	**83.8%**
JAPVOW	–	–	**88.9%**	**92.3%**	28.3%	28.5%	69.7%

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

Moreover, we already noticed that both IHMM-KNN and IHMM-BHATT have a precise counterpart obtained by setting $s = 0$ in the constraints of the imprecise Dirichlet model. A comparison with these precise classifiers is in Table 5, where, besides the accuracy of the precise methods on the whole benchmark, also the accuracy evaluated only on the determinate and indeterminate instances is reported. The pattern *p-set-acc* < *acc* < *p-sing-acc* is always satisfied by both the methods. These credal classifiers are therefore effective in discriminating hard-to-classify from “easy” instances. In other words, if the credal classifier returns a single class label, we can reasonably assume 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 very likely to be wrong.

8. Conclusions and outlooks

Two new credal classifiers for temporal data have been presented. Imprecise HMMs are learned from each sequence. The first classifier summarizes the model

Dataset	IHMM-KNN			IHMM-BHATT		
	p-sing-acc	*p-set-acc*	*acc*	*p-sing-acc*	*p-set-acc*	*acc*
KTH$_1$	100.0%	84.2%	88.0%	81.2%	59.1%	78.0%
KTH$_2$	100.0%	43.2%	46.7%	60.0%	41.2%	50.0%
KTH$_3$	94.1%	68.9%	71.8%	75.0%	52.6%	69.3%
KTH$_4$	100.0%	69.2%	70.6%	78.2%	57.7%	74.7%
KTH	100.0%	68.1%	71.3%	78.0%	54.3%	68.0%
Weizmann	100.0%	75.5%	81.9%	78.6%	49.8%	69.3%
AUSLAN	–	–	–	85.2%	45.5%	72.2%
JAPVOW	–	–	–	99.6%	81.1%	95.1%

Table 5: Precise single and set accuracy of iHMM-kNN. The same classifier with $s = 0$ is used as a precise counterpart and its accuracy is in the last column. The values of *p-sing-acc* in this table coincide therefore with the *sing-acc* in Table 3.

with a hyperbox in the feature space. This datum is finally classified by a generalization of the k-NN approach. The second classifier uses an interval-valued dissimilarity measure. The results are promising: the algorithms outperform a credal classifier previously proposed for this task and compete with the state-of-the-art methods. As a future work, we want to investigate novel, more reliable, learning techniques like for instance the likelihood-based approach already considered for complete data in Antonucci et al. (2011a). Also more complex topologies should be considered.

Appendix A. Solving the Linearly Constrained Quadratic Optimization

Let us consider the linearly constrained quadratic optimization tasks to be solved by the IHMM-BHATT algorithm. These task can be solved in polynomial (roughly cubic) time because the solution lies on an extreme point of the feasible region (Boyd and Vandenberghe, 2004). The minimization of the objective function in Equation (24) can be rewritten 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 $\tilde{P}(X) \in \tilde{K}(X)$. Its solution, which clearly coincides with the optimal solution $\tilde{P}^*(X)$ is known to belong to $\mathrm{ext}[\tilde{K}(X)]$ being the solution of the linear problem. We can similarly prove that $\tilde{Q}^*(X') \in \mathrm{ext}[\tilde{K}(X')]$, and the results is the same 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 $h_1 := I_{x'}$[16], we obtain:

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

and similarly for the upper.

References

Antonucci, A., 2012. An interval-valued dissimilarity measure for belief functions based on credal semantics, in: Denoeux, T., Masson, M. (Eds.), Belief Functions: Theory and Applications - Proceedings of the 2nd International Conference on Belief Functions, Springer. pp. 37–44.

Antonucci, A., Cattaneo, M., Corani, G., 2011a. Likelihood-based naive credal classifier, in: ISIPTA '11: Proceedings of the Seventh International Symposium on Imprecise Probability: Theories and Applications, SIPTA. pp. 21–30.

Antonucci, A., Cattaneo, M., Corani, G., 2012. Likelihood-based robust classification with Bayesian networks, in: Communications in Computer and Information Science, Springer Berlin / Heidelberg. pp. 491–500.

Antonucci, A., de Rosa, R., Giusti, A., 2011b. 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. pp. 474–478.

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

[16] 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.

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

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

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

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

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

de Cooman, G., Zaffalon, M., 2004. Updating beliefs with incomplete observations. Artificial Intelligence 159, 75–125.

Corani, G., Antonucci, A., 2012. Credal Ensembles of Classifiers. Computational Statistics and Data Analysis .

Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the em algorithm. Journal of the Royal Statistical Society, Series B 39, 1–38.

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

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

Kies, J., 1997. Empirical Methods for Evaluating Video-Mediated Collaborative Work. Ph.D. thesis. Virginia Tech.

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

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

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

Schuldt, C., Laptev, I., Caputo, B., 2004. Recognizing human actions: a local SVM approach, in: Proc. of International Conference on Pattern Recognition.

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

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

Ten Holt, G.A., Reinders, M.J.T., Hendriks, E., 2007. Multi-dimensional dynamic time warping for gesture recognition. Time 5249, 23–32.

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

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

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

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

Zaffalon, M., 2002. The naive credal classifier. J. Stat. Plann. Inference 105, 5–21.

Zaffalon, M., Corani, G., Mauá, D., 2011. Utility-based accuracy measures to empirically evaluate credal classifiers, in: ISIPTA '11: Proceedings of the Seventh International Symposium on Imprecise Probability: Theories and Applications, SIPTA. pp. 401–410.

Zaffalon, M., Fagiuoli, E., 2003. Tree-based credal networks for classification. Reliable Computing 9, 487–509.