Unsupervised classification using hidden Markov chain with unknown noise copulas and margins

Stéphane Derrode[a,*], Wojciech Pieczynski[b]

[a] *École Centrale de Lyon, LIRIS, CNRS UMR 5205, Écully, France.*
[b] *Telecom Sudparis, SAMOVAR, CNRS UMR 5157, Évry, France.*

Abstract

We consider the problem of unsupervised classification of hidden Markov models (HMC) with dependent noise. Time is discrete, the hidden process takes its values in a finite set of classes, while the observed process is continuous. We adopt an extended HMC model in which the rich possibilities of different kinds of dependence in the noise are modelled via copulas. A general model identification algorithm, in which different noise margins and copulas corresponding to different classes are selected in given families and estimated in an automated way, from the sole observed process, is proposed. The interest of the whole procedure is shown via experiments on simulated data and on a real SAR image.

Keywords: Hidden Markov models, Dependent noise, Model selection, Iterative conditional estimation, Copulas, Unsupervised classification, Pearson's system of distributions.

1. Introduction

The paper deals with the problem of unsupervised estimation of a hidden discrete process $\boldsymbol{X}_1^N = (X_1, \ldots, X_N)$ from an observed continuous one $\boldsymbol{Y}_1^N = (Y_1, \ldots, Y_N)$. Hidden Markov models (HMMs) are very widely used to deal with the problem. Indeed, they allow recursive computations of different quantities used in optimal Bayesian processing in linear time. There are many papers

[*] Corresponding author

following the pioneering ones [1, 2], dealing with various application areas. Let us mention some recent general papers or books about general setting [3, 4, 5], signal and image processing [3], economy and finance [6, 7], or biology [8, 9]. Besides, copulas [10, 11] are also of interest in numerous situations, due to their ability of modelling dependent non-Gaussian data [12, 13, 14, 15]. Their use goes increasing in different areas. Mainly applied in economy and finance [16, 17, 18, 19, 20, 21], they are becoming increasingly used in other fields, such as in signal or image processing processing [22, 23, 24, 25] or in ecology [26, 27, 28].

However, despite their great benefit when used separately, there is very little research and applications that combines them. First papers on the subject date from about ten years: copulas use has been introduced at temporal level in hidden Markov chains with dependent noise (HMC-DN) in [29], at vectorial level in hidden Markov chains in [30], and in hidden Markov trees in [31]. Some applications using vectorial-level copulas have been proposed in the context of hidden Markov chains [32], hidden Markov trees [33], hidden Markov fields [34, 35], or general Bayesian networks [36]. They were showed to be especially useful in multi-sensor image processing where sensors are dependent and not Gaussian [34, 35]. Temporal-level copulas remain, for their part, very little used. This is certainly due to the fact that the observations in HMMs are usually assumed to be independent conditionally on the hidden data, and thus there is no dependency to model. However, taking into account the noise dependence is of interest, and using the right copulas can have strong influence on the efficiency of Bayesian processing methods in HMMs with correlated noise [37].

Our paper deals with the problem of unsupervised classification of hidden Markov chains with copulas used at temporal level. The novelty of the work is to propose a general method allowing one to search the best copulas in a finite set of admissible copulas, as well as the best margins in a finite set of admissible margins. In addition, the admissible sets of copulas and margins can vary with the hidden discrete data. This allows one to select, from the only observed data, the best model in a quite rich set of possible models. Therefore we simultaneously extend, first, the method presented in [37] where the copulas where searched while the forms of margins were assumed known and, second, the method presented in [38, 39] where the margins were searched while assuming independence.

Let us notice that the presented results can be almost directly applied to more complex models than the HMC-DNs considered. Indeed, when parameter estimation is concerned, dealing with “pairwise Markov models” (PMMs) [40, 41] or even “triplet Markov models” (TMMs), –which includes non stationary PMMs [42], hidden semi-Markov models [43], or still hidden bivariate Markov models [44]–, is a quite similar problem [42, 43].

The organization of the paper is the following. In next Section we recall the basics about HMM and how a dependent noise can be modelled using a copula representation. The general model identification method we propose is then specified in Section three. Section four is devoted to recall the classic computations in HMM-DN for different quantities of interest. Fifth section contains some systematic experiments and the segmentation result of a real SAR image. The last Section draws conclusions and proposes a few perspectives.

2. HMM with dependent noise and copulas

Let us consider two random sequences $\boldsymbol{X}_1^N = (X_1, \ldots, X_N)$ and $\boldsymbol{Y}_1^N = (Y_1, \ldots, Y_N)$, taking their values in $\Omega = \{1, \ldots, K\}$ and $\mathbb{R}$ respectively. $\boldsymbol{X}_1^N$ is hidden, while $\boldsymbol{Y}_1^N$ is observed, and the problem is to estimate $\boldsymbol{X}_1^N$ from $\boldsymbol{Y}_1^N$. Optimal Bayesian methods can be used for the classic hidden Markov models (HMMs), whose distribution is defined with

$$p\left(\boldsymbol{x}_1^N, \boldsymbol{y}_1^N\right) = p\left(x_1\right) p\left(y_1 \left| x_1\right.\right) p\left(x_2 \left| x_1\right.\right) p\left(y_2 \left| x_2\right.\right) \ldots p\left(x_N \left| x_{N-1}\right.\right) p\left(y_N \left| x_N\right.\right). \tag{1}$$

HMMs can also be defined as verifying two hypotheses:

$$\boldsymbol{X}_1^N \text{ is Markov;} \tag{2}$$

$$p\left(\boldsymbol{y}_1^N \left| \boldsymbol{x}_1^N\right.\right) = \prod_{n=1}^{N} p\left(y_n \left| x_n\right.\right). \tag{3}$$

Let us notice that (3) means that the random variables $Y_1, \dots, Y_N$ are independent conditionally on $\boldsymbol{X}_1^N$; for this reason we will call the classic HMM (2)-(3) “HMM with independent noise” (HMM-IN).

It is possible to consider more general models in which both processes $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$ and $\boldsymbol{X}_1^N$ are Markov and in which the same Bayesian processing as in HMM-IN remains possible. The distribution of such models is written

$$p\left(x_{n+1}, y_{n+1} \left| x_n, y_n \right.\right) = p\left(x_{n+1} \left| x_n \right.\right) p\left(y_{n+1} \left| x_n, y_n, x_{n+1} \right.\right). \tag{4}$$

In this kind of models, called HMM with dependent noise (HMM-DN) Y_1, ..., Y_N are (possibly) dependent conditionally on $\boldsymbol{X}_1^N$. Thus an HMM-IN is an HMM-DN for which $p\left(y_{n+1} \left| x_n, y_n, x_{n+1} \right.\right) = p\left(y_{n+1} \left| x_{n+1} \right.\right)$.

Remark 2.1 It has been shown in [41, 40] that the Markovianity of $\boldsymbol{X}_1^N$ is not even required, and the following model called “pairwise Markov model” (PMM)

$$p\left(\boldsymbol{x}_1^N, \boldsymbol{y}_1^N\right) = p\left(x_1, y_1\right) \sum_{n=1}^{N-1} p\left(x_{n+1}, y_{n+1} \left| x_n, y_n \right.\right) \tag{5}$$

allows the same processing than HMM-DNs.

In this paper we will deal with the stationary reversible case, which means that $p\left(x_n, y_n, x_{n+1}, y_{n+1}\right)$ does not depend on $n = 1, \dots, N-1$, and the distributions $p\left(x_{n+1}, y_{n+1} \left| x_n, y_n \right.\right)$ and $p\left(x_n, y_n \left| x_{n+1}, y_{n+1} \right.\right)$ are equal. In that case, an HMM-DN is a particular case of PMM for which we have

$$p\left(y_{n+1} \left| x_{n+1}, x_n \right.\right) = p\left(y_{n+1} \left| x_{n+1} \right.\right), \tag{6}$$

for all $n \in [1, N-1]$, see [41]. Thus in the model considered in this paper we have simultaneously (4) and (6). Let us notice that (6) does not imply that $p\left(y_{n+1} \left| x_n, y_n, x_{n+1} \right.\right)$ can be reduced to a simpler expression: the ditribution of Y_{n+1} conditional on X_n, Y_n, X_{n+1} can depend on the three variables.

The distribution of such a stationary reversible HMM-DN $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$ is defined by

$$p\left(x_1, y_1, x_2, y_2\right) = p\left(x_1, x_2\right) p\left(y_1, y_2 \left| x_1, x_2 \right.\right). \tag{7}$$

The aim of this paper is to consider $p\left(y_1, y_2 \left| x_1, x_2 \right.\right)$ in (7) under very general form and to propose a way for its estimation, together with $p\left(x_1, x_2\right)$, from the observed sequence $\boldsymbol{Y}_1^N$. More precisely, for given (x_1, x_2), $p\left(y_1, y_2 \left| x_1, x_2 \right.\right)$ is defined by

- two margins $p\left(y_1 \left| x_1, x_2 \right.\right) = p\left(y_1 \left| x_1 \right.\right) = f_{x_1}^l(y_1)$ and $p\left(y_2 \left| x_1, x_2 \right.\right) = p\left(y_2 \left| x_2 \right.\right) = f_{x_2}^r(y_2)$, according to (6) ($l$ and r stand for 'left' and 'right' to distinguish between the left and right variables, see below);
- a copula C with pdf $c\left(F_{x_1}^l(y_1), F_{x_2}^r(y_2) \left| x_1, x_2 \right.\right) = c_{x_1,x_2}(F_{x_1}^l(y_1), F_{x_2}^r(y_2))$, where F is the cumulative distribution function (cdf) corresponding to f.

We recall that a copula C is defined as a cumulative distribution function on $[0, 1]^2$ such that the corresponding marginal cumulative functions are identity, which also means that the corresponding marginal distributions on $[0, 1]$ are uniform distributions, see *e.g.* [10]. Let $h(y_1, y_2)$ be a probability distribution on $\mathbb{R}^2$, which will be assumed continuous in this paper. Let $H(y_1, y_2)$ be the corresponding cumulative function, $h^l(y_1)$ and $h^r(y_2)$ the corresponding marginal densities, and $H^l(y_1)$, $H^r(y_2)$ the associated cumulative functions. According to Sklar's theorem [11] there exists an unique copula C such that

$$H(y_1, y_2) = C(H^l(y_1), H^r(y_2)). \tag{8}$$

Setting $c(u, v) = \frac{\partial\partial C(u,v)}{\partial u \partial v}$ and deriving (8) with respect to y_1, y_2 gives

$$h(y_1, y_2) = h^l(y_1)h^r(y_2)\ c(H^l(y_1), H^r(y_2)). \tag{9}$$

Thus any continuous probability distribution $h(y_1, y_2)$ is given by a triplet h^l, h^r, and a probability distribution c on $[0, 1]^2$ with uniform margins. Conversely, such a triplet defines a probability distribution on $[0, 1]^2$ with (9). Such a representation of $h(y_1, y_2)$ is of interest as every distribution among h^l, h^r, c can be modified independently from the two others. For example, a Gaussian copula $h(y_1, y_2)$ is given by Gaussian margins h^l, h^r, and a Gaussian copula c. Replacing in (8) c with another non Gaussian copula c' we obtain a non Gaussian distribution $H'(y_1, y_2)$ with Gaussian margins. We can also keep the

Gaussian copula c and replace the Gaussian margins by any other ones. This offers a very rich set of possibilities easy to handle with.

We will assume that for each $(x_1, x_2) \in \Omega^2$, each $f^l_{x_1}$ and each $f^r_{x_2}$ belongs to a parametric set of distributions, which themselves belongs to a finite family of parametric sets of distributions. For example, imagine that f^l_1 can be Gaussian or Gamma, f^r_1 can be Beta, Gamma or Rayleigh, f^l_2 can be Beta or Gamma, f^r_2 can be exponential and so on for $x_1 = 3, \ldots, K$, $x_2 = 3, \ldots, K$. Thus, for each $(x_1 = i, x_2 = j)$, we have to find what is the general form of the distributions f^l_i and f^r_j, and we have to find the parameters, which precisely define the distribution of the determined shape. Similarly, for each $(x_1 = i, x_2 = j)$ we have to find general form of copula $c_{i,j}$ and estimate the parameters, which set the copula in the set of copulas having the same form. For example, $c_{1,1}$ can be Gumbel or Gaussian, $c_{1,2}$ can be Gaussian or Clayton, $c_{2,1}$ can be Student, product or Cubic Section, $c_{1,3}$ can be A14 or Clayton, and so on for $x_1 = 3, \ldots, K$, $x_2 = 3, \ldots, K$.

As mentionned above, such problems have been partly dealt with in [37]. Margins have been considered as known, and copulas have been searched - for each (x_1, x_2) - in finite sets of possible copulas. In particular, it has been showed that the right form of copula - for each (x_1, x_2) - was of importance for the efficiency of classification. Besides, automatic choice of the form of margins $p\left(y_1 \left| x_1 \right.\right)$ for each x_1 - in the HMM-INs case - have been studied in [38, 39] and it has also been showed that the use of right forms of margins was of importance. Thus in this paper we address these two problems simultaneously, which results in a very general method of model identification from hte only observations $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$.

3. Shapes and parameters estimation

So, the distribution of a stationary reversible PMM is defined by $p\left(x_1, x_2\right)$ and $p\left(y_1, y_2 \left| x_1, x_2 \right.\right) = f_{x_1,x_2}(y_1, y_2)$, the latter being defined by margins $f^l_{x_1}$, $f^r_{x_2}$, such that when $x_1 = x_2$ we have $f^l_{x_1} = f^r_{x_2}$, and a copula c_{x_1,x_2}, such that $c_{x_1,x_2} = c_{x_2,x_1}$, for each x_1, x_2 in Ω. The problem we deal with is to find $p\left(x_1, x_2\right)$ and $p\left(y_1, y_2 \left| x_1, x_2\right.\right)$ (for each $(x_1, x_2) = (i, j)$), using the sole observation $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$.

Let us concentrate on the search of $f_{i,j}(y_1, y_2)$, which is thus defined by f_i^l, f_j^r, and $c_{i,j}$. The problem is twofold:

1. What forms these three functions are of?
2. Once the form known, what are the related parameters?

We are going to deal with these two problems simultaneously in a very wide-ranging setting.

As $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$ is stationary reversible, only K different margins f_i are required to define an HMM-DN model (as for HMM-IN). For each $i \in \Omega = \{1, \ldots, K\}$ the form of f_i is not know, but it belongs to a known set of possible shapes $\mathcal{F}_i = \left\{\mathsf{F}_{i,1}, \ldots, \mathsf{F}_{i,K(i)}\right\}$. Besides, each form $\mathsf{F}_{i,k}$ is a parametric set of probability distributions $\mathsf{F}_{i,k} = \left\{f_{\theta(i,k)}\right\}_{\theta(i,k) \in \Theta(i,k)}$. Similarly, for each $i, j \in \Omega$, $c_{i,j}$ is not known, but it belongs to a known set of possible forms $\mathcal{G}_{i,j} = \left\{\mathsf{G}_{i,j,1}, \ldots, \mathsf{G}_{i,j,M(i,j)}\right\}$, and each of them is a parametric set $\mathsf{G}_{i,j,m}$ of copulas $\mathsf{G}_{i,j,m} = \left\{c_{\alpha(i,j,k)}\right\}_{\alpha(i,j,m) \in A(i,j,m)}$. Finally, for each $i, j \in \Omega$, the problem is to find from $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$:

1. the right forms $\mathsf{F}_{i,k}$ and $\mathsf{G}_{i,j,m}$;
2. the right parameters $\theta(i,k)$ and $\alpha(i,j,m)$.

Besides, we assume to have two families of estimators. First, for each $i \in \Omega$ and each $k \in \{1, \ldots, K(i)\}$, there exists an estimator $\hat{\theta}(i,k)(\boldsymbol{y}_1^{\star N})$ giving $\theta(i,k)$ from $\boldsymbol{Y}_1^{\star N}$ whose distribution is such that the marginal distributions $p\left(y_n^\star\right)$ are equal and belong to $\mathsf{F}_{i,k}$. Second, for each $i, j \in \Omega$ and each $m \in \{1, \ldots, M(i,j)\}$ there exists an estimator $\hat{\alpha}(i,j,m)(\boldsymbol{y}_1^{\star N})$ giving $\alpha(i,j,m)$ from $\boldsymbol{Y}_1^{\star N}$ whose distribution is such that the distributions $p\left(y_n^\star, y_{n+1}^\star\right)$ are equal and belong to $\mathsf{G}_{i,j,m}$. Let us notice that these conditions are not strong. Indeed, the problem is to find right shapes and right parameters without knowing realizations of $\boldsymbol{X}_1^N = \boldsymbol{x}_1^N$, and thus assuming that we can solve the problem by knowing them is the least we should assume.

We will also assume, for each $i, j \in \Omega$, to have two "decision rules" $\mathcal{D}^1$ and $\mathcal{D}^2$ allowing to perform, from realizations $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$, the following decisions:

- For any $f_{\theta(i,1)} \in \mathsf{F}_{i,1}, \ldots, f_{\theta(i,K(i))} \in F_{i,K(i)}$, $\mathcal{D}^1$ makes correspond to $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$ an unique element in $\left\{f_{\theta(i,1)}, \ldots, f_{\theta(i,K(i))}\right\}$.

- For any $c_{\alpha(i,j,1)} \in G_{i,j,1}, \ldots, c_{\alpha(i,j,M(i,j))} \in G_{i,j,M(i,j)}$, $\mathcal{D}^2$ makes correspond to $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$ an unique element in $\left\{c_{\alpha(i,j,1)}, \ldots, c_{\alpha(i,j,M(i,j))}\right\}$.

Finally, we observe a sample $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$ of a stationary reversible HMC $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$ and the problem is to estimate its distribution in the frame described above. Thus we have to find, for each $i, j \in \Omega$:

1. $p_{i,j} = p\left(x_1 = i, x_2 = j\right)$;
2. $k \in \{1, \ldots, K(i)\}$ - which gives $\mathsf{F}_{i,k}$ in $\mathcal{F}_i = \left\{\mathsf{F}_{i,1}, \ldots, \mathsf{F}_{i,K(i)}\right\}$ -, and $\theta(i,k)$ in $\Theta(i,k)$, which gives $f_i = f_{\theta(i,k)}$ in $\mathsf{F}_{i,k}$;
3. $m \in \{1, \ldots, M(i,j)\}$ - which gives $\mathsf{G}_{i,j,m}$ in $\mathcal{G}_{i,j} = \left\{\mathsf{G}_{i,j,1}, \ldots, \mathsf{G}_{i,j,M(i,j)}\right\}$ -, and $\alpha(i,j,k)$ in $A(i,j,k)$, which gives $c_{i,j} = c_{\alpha(i,j,m)}$ in $\mathsf{G}_{i,j,m}$.

The general idea of the iterative GICE, drawn from the idea of the simple ICE, is the following. At a given iteration one uses the current shapes and parameters to sample a sequence $\boldsymbol{x}_1^N$ according to the distribution of $\boldsymbol{X}_1^N$ conditional on $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$, and this sequence is dealt with as if it were a true realization of $\boldsymbol{X}_1^N$. Then for each possible shape one uses the sampled sequence (with $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$) to estimate the corresponding parameters, which fix the possible shapes. Finally, the decision rules $\mathcal{D}^1$ and $\mathcal{D}^2$ allow one to determine, from $\boldsymbol{x}_1^N$ and $\boldsymbol{y}_1^N$, the shapes (with the corresponding parameters just fixed by estimators) which will be kept for the next iteration.

We will need the following definition. Let $(\boldsymbol{x}_1^N, \boldsymbol{y}_1^N)$ be a realization of a stationary reversible HMC-DN $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$. We will denote by $\boldsymbol{y}_{i,j}(\boldsymbol{x}_1^N)$ the sequence of all couples (y_n, y_{n+1}) in $\boldsymbol{y}_1^N$ such that $(x_n, x_{n+1}) = (i,j)$, and by $\boldsymbol{y}^i(\boldsymbol{x}_1^N)$ the sequence of y_n in $\boldsymbol{y}_1^N$ such that $x_n = i$. In other words

$$\begin{aligned}
\boldsymbol{y}_{i,j}(\boldsymbol{x}_1^N) &= \left\{(y_n, y_{n+1}) \subset \boldsymbol{y}_1^N | (x_n, x_{n+1}) = (i,j)\right\}, \\
\boldsymbol{y}^i(\boldsymbol{x}_1^N) &= \left\{y_n \subset \boldsymbol{y}_1^N | x_n = i\right\}.
\end{aligned}$$

The "generalized iterative conditional estimation" (GICE) we propose to search $(p_{i,j}, f_i, c_{i,j})$, for each $i, j \in \Omega$, is the following iterative method:

1. Initialize GICE with $(p^0_{i,j}, f^0_i, c^0_{i,j})$ found with a simple method;
2. For each $i, j \in \Omega$, to find $(p^{q+1}_{i,j}, f^{q+1}_i, c^{q+1}_{i,j})$ from $(p^q_{i,j}, f^q_i, c^q_{i,j})$ and $\boldsymbol{y}^N_1$:
   (a) set $p^{q+1}_{i,j} = \frac{1}{N-1}\sum_{n=1}^{N-1} p^q(x_n, x_{n+1}|\boldsymbol{y}^N_1)$, where $p^q(x_n, x_{n+1}|\boldsymbol{y}^N_1)$ are based on $(p^q_{i,j}, f^q_i, c^q_{i,j})$, see Section 4.2 for their computation;
   (b) sample $(\boldsymbol{x}^N_1)^{q+1}$ according to $p\left(\boldsymbol{x}^N_1 \left|\boldsymbol{y}^N_1\right.\right)$ based on $(p^q_{i,j}, f^q_i, c^q_{i,j})$, see Section 4.1 for the sampling method;
   (c) for each $i, j \in \Omega$, each $k \in \{1, \ldots, K(i)\}$, and each $m \in \{1, \ldots, M(i, j)\}$, consider $\theta^{q+1}(i, k) = \hat{\theta}(i, k)(\boldsymbol{y}^i((\boldsymbol{x}^N_1)^{q+1}))$ and $\alpha^{q+1}(i, j, m) = \hat{\alpha}(i, j, m)(\boldsymbol{y}_{i,j}((\boldsymbol{x}^N_1)^{q+1}))$;
   (d) use $\mathcal{D}^1$ and $\boldsymbol{y}^i((\boldsymbol{x}^N_1)^{q+1}))$ to determine the unique element f^{q+1}_i in $\left\{f_{\theta^{q+1}(i,1)}, \ldots, f_{\theta^{q+1}(i,K(i))}\right\}$, and use $\mathcal{D}^2$ and $\boldsymbol{y}_{i,j}((\boldsymbol{x}^N_1)^{q+1})$ to determine the unique element $c^{q+1}_{i,j}$ in $\left\{c_{\alpha^{q+1}(i,j,1)}, \ldots, c_{\alpha^{q+1}(i,j,M(i,j))}\right\}$.
3. Stop according to some criterion.

Such a general method offers rich possibilities of particular algorithms. In fact, there exist, in general, different estimators $\hat{\theta}(i, k)$, $\hat{\alpha}(i, j, m)$. Similarly, there exists a great deal of different decision rules $\mathcal{D}^1$ and $\mathcal{D}^2$.

Remark 3.1 GICE is an extension, containing margin's and copula's automated selection, of the classical ICE method [38, 40, 39, 45]. Let us briefly recall how the latter runs and what are its differences with the well-known "expectation-maximization" (EM) method [1, 46, 2]. Let us consider two random processes $(\boldsymbol{V}, \boldsymbol{Y})$ whose distribution depends on a vector of parameters $\boldsymbol{\theta} = \{\theta_1, \ldots, \theta_m\}$. The problem is to estimate $\boldsymbol{\theta}$ from $\boldsymbol{Y}$. The ICE method is an iterative method based on the following principle. Let $\hat{\boldsymbol{\theta}}(\boldsymbol{v}, \boldsymbol{y})$ be an estimator of $\boldsymbol{\theta}$ from complete data $(\boldsymbol{V}, \boldsymbol{Y}) = (\boldsymbol{v}, \boldsymbol{y})$ and let us assume that we can sample realizations of $\boldsymbol{V}$ according to $p\left(\boldsymbol{v}\left|\boldsymbol{y}\right.\right)$. The ICE sequence is obtained as follows:

1. Initialize $\boldsymbol{\theta}^0$;
2. Compute $\boldsymbol{\theta}^{q+1} = E\left[\hat{\boldsymbol{\theta}}(\boldsymbol{V}, \boldsymbol{Y})\left|\boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{\theta}^q\right.\right]$. In practice, $\theta^{q+1}_i = E\left[\hat{\theta}_i(\boldsymbol{V}, \boldsymbol{Y})\left|\boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{\theta}^q\right.\right]$ is computed for the components θ_i for which this computation can be

carried on explicitly and for the remaining components one simulates $\boldsymbol{v}_1^q, \ldots, \boldsymbol{v}_l^q$ according to $p\left(\boldsymbol{v}\,|\boldsymbol{y}, \boldsymbol{\theta}^q\right)$ and one sets $\theta_i^{q+1} = \left[\hat{\theta}_i(\boldsymbol{v}_1^q, \boldsymbol{y}) + \ldots + \hat{\theta}_i(\boldsymbol{v}_l^q, \boldsymbol{y})\right]/l$, which approximates the expectation.

In practise one takes often $l = 1$, which is done in GICE. We see that ICE is applicable under two very mild hypotheses: existence of an estimator $\hat{\boldsymbol{\theta}}(\boldsymbol{v}, \boldsymbol{y})$ from the complete data, and the ability of simulating $\boldsymbol{V}$ according to $p\left(\boldsymbol{v}\,|\boldsymbol{y}\right)$.

The principle of EM is

1. Initialize $\boldsymbol{\theta}^0$;
2. Compute -or approximate- $\theta^{q+1} = \arg\max_{\boldsymbol{\theta}} E\left[\ln\left(p_{\boldsymbol{\theta}}(\boldsymbol{V}, \boldsymbol{Y})\right)|\boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{\theta}^q\right]$, where $p_{\boldsymbol{\theta}}$ is a likelihood.

One can see that ICE is simpler to use than EM as there is no maximization step. When $\hat{\boldsymbol{\theta}}(\boldsymbol{v}, \boldsymbol{y})$ used in ICE is the Maximum Likelihood estimator we have $\theta^{q+1} = E\left[\arg\max_{\boldsymbol{\theta}} \ln\left(p_{\boldsymbol{\theta}}(\boldsymbol{V}, \boldsymbol{Y})\right)|\boldsymbol{Y} = \boldsymbol{y}, \boldsymbol{\theta}^q\right]$; ICE and EM can give the same sequence when "expectation" and "maximization" commute, which occurs, roughly speaking, in exponential models [46].

Example 3.1 One possible rule $\mathcal{D}^1$ for choosing among K densities $f_1, \ldots, f_K$ from $\boldsymbol{Y}_1^r = \boldsymbol{y}_1^r$, successfully used in [39] to search the margins forms in classic multisensor HMM-IN, is the minimization of the Kolmogorov distance d between these distributions and the empirical distribution. Let $F_1, \ldots, F_K$ be the related cumulative distribution functions and $F_e(y) = \frac{1}{r}\sum_{n=1}^{r} 1_{[y_n < y]}$ the empirical cdf. We have

$$\mathcal{D}^1(\boldsymbol{y}_1^r) = \arg \inf_{F^k \in \{F_1, \ldots, F_K\}} \left[d\left(F_k, F_e\right)\right], \tag{10}$$

with the Kolmogorov distance d between two cdfs F and F' given by $d(F, F') = \sup_{y \in \mathbb{R}} |F(y) - F'(y)|$. We may notice that this distance is quite easy to compute, as the sup has to be searched only on $y_1, \ldots, y_r$.

Example 3.2 In some situations in which there exist estimators $\hat{\theta}(i)$ such that the estimated parameter $\hat{\theta}(i)(\boldsymbol{y}_1^N)$ also gives the form $\mathsf{F}_{i,k}$ in $\mathcal{F}_i = \left\{\mathsf{F}_{i,1}, \ldots, \mathsf{F}_{i,K(i)}\right\}$. This is the case when $\mathcal{F}_{i,j}$ belongs to the Pearson's system of distributions [47]. In such case, which has been successfully used in the context of independent noise and hidden multi-sensor Markov fields in [38], there is no rule $\mathcal{D}^1$ to use as the choice of the shape and the estimation of its parameters are performed simultaneously, *cf.* Section 5.

Example 3.3 One possible rule $\mathcal{D}^2$ for choosing among M copulas $c_1, \ldots, c_M$ from (y_1, y_2), (y_2, y_3), $\ldots$, (y_{r-1}, y_r) that we will call "pseudo-likelihood maximization" (PLM) method and which will be used in experiments below, is the following:

$$\mathcal{D}^2((y_1, y_2), (y_2, y_3), \ldots, (y_{r-1}, y_r)) = \arg \sup_{c_m \in \{c_1, \ldots, c_M\}} \prod_{i=2}^{r} c_m(y_{i-1}, y_i). \quad (11)$$

We also tested the "Bayesian copula selection method" proposed in [48] and considered in [37], and the latter turns out to be less efficient than PLM in the context of the experiments considered in Section 5.

4. Sampling and classification of HMM-DNs

We recall in this section the classic computations needed in GICE and in Bayesian MPM classification. Let us consider a reversible stationary HMM-DN $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$, with the distribution defined by $p\left(x_1, y_1\right)$ and $p\left(x_2, y_2 \left| x_1, y_1 \right.\right)$.

4.1. Sampling HMM-DNs

To sample realizations of $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$, we need $p\left(x_1, y_1\right)$ and $p\left(x_2, y_2 \left| x_1, y_1 \right.\right)$ (equal to $p\left(x_{n+1}, y_{n+1} \left| x_n, y_n \right.\right)$ for each $n = 2, \ldots, N-1$). Adopting the notations of previous Section, for each $i, j \in \Omega$, let $p_{i,j} = p\left(x_1 = i, x_2 = j\right)$ and $p\left(y_1, y_2 \left| x_1 = i, x_2 = j \right.\right) = f_{i,j}(y_1, y_2) = f_i^l(y_1) f_j^r(y_2)\ c_{i,j}(F_i^l(y_1), F_j^r(y_2))$. In addition, let $p_i = p\left(x_1 = i\right) = \sum_{j=1}^{K} p_{i,j}$ and $p_{j|i} = p\left(x_2 = j \left| x_1 = i \right.\right)$. Then $p\left(x_1, y_1\right) = p\left(x_1\right) p\left(y_1 \left| x_1 \right.\right)$ and $p\left(x_2, y_2 \left| x_1, y_1 \right.\right) = p\left(x_2 \left| x_1 \right.\right) p\left(y_2 \left| x_1, y_1, x_2 \right.\right)$, see eq. (4), are given by

$$p\left(x_1 = i\right) = p_i, \quad p\left(y_1 \left| x_1 = i \right.\right) = f_i^l(y_1), \quad p\left(x_2 = j \left| x_1 = i \right.\right) = p_{j|i},$$

and

$$\begin{aligned} p\left(y_2 \left| x_1 = i, y_1, x_2 = j \right.\right) &= \frac{p\left(y_1, y_2 \left| x_1 = i, x_2 = j \right.\right)}{p\left(y_1 \left| x_1 = i \right.\right)} \\ &= f_j^r(y_2)\ c_{i,j}(F_i^l(y_1), F_j^r(y_2)), \end{aligned}$$

So that we finally get

$$p\left(x_2 = j, y_2 \left| x_1 = i, y_1 \right.\right) = p_{j|i}\; f_j^r(y_2)\; c_{i,j}(F_i^l(y_1), F_j^r(y_2)). \tag{12}$$

There are different methods for sampling $Y_1 = y_1$ and $Y_2 = y_2$ according to (8)-(9); in particular, acceptance-rejection method [49] may be used.

4.2. Estimation $\boldsymbol{X}_1^N$ *in HMM-DN* $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$

Let us recall how the distributions $p\left(x_n \left| \boldsymbol{y}_1^N \right.\right)$, $p\left(x_n, x_{n+1} \left| \boldsymbol{y}_1^N \right.\right)$, and $p\left(x_{n+1} \left| x_n, \boldsymbol{y}_1^N \right.\right)$ are computed in an HMM-DN $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$. The first one is used in Bayesian Maximum Posterior Mode (MPM) classification, which consists of estimating $\boldsymbol{X}_1^N = (X_1, \ldots, X_N) = (x_1, \ldots, x_N)$ by $\hat{\boldsymbol{x}}_1^N = (\hat{x}_1, \ldots, \hat{x}_N)$ such that each $\hat{x}_n$ maximizes $p\left(x_n \left| \boldsymbol{y}_1^N \right.\right)$, and which minimizes the mean rate of errors. The second and third ones are used in points (a) and (b) of GICE algorithm, respectively.

Classically, we have

$$p\left(x_n, \boldsymbol{y}_1^N\right) = p\left(x_n, \boldsymbol{y}_1^n\right) p\left(\boldsymbol{y}_{n+1}^N \left| x_n, y_n \right.\right) = \alpha_n(x_n)\beta_n(x_n), \tag{13}$$

where α_n and β_n are called "forward" and "backward" probabilities. They can be computed with the following "forward" and "backward" recursions:

$$\begin{aligned}
\alpha_1(x_1) &= p\left(x_1, y_1\right); && (14)\\
\alpha_{n+1}(x_{n+1}) &= \sum_{x_n} p\left(x_{n+1}, y_{n+1} \left| x_n, y_n \right.\right) \alpha_n(x_n), &&\\
&\qquad\qquad \text{for } n = 1, \ldots, N-1; && (15)\\
\beta_N(x_N) &= 1; && (16)\\
\beta_n(x_n) &= \sum_{x_{n+1}} p\left(x_{n+1}, y_{n+1} \left| x_n, y_n \right.\right) \beta_{n+1}(x_{n+1}), &&\\
&\qquad\qquad \text{for } n = N-1, \ldots, 1. && (17)
\end{aligned}$$

So that we can write

$$\begin{aligned}
p\left(x_n \left| \boldsymbol{y}_1^N \right.\right) &\propto \alpha_n(x_n)\beta_n(x_n),\\
p\left(x_n, x_{n+1} \left| \boldsymbol{y}_1^N \right.\right) &\propto \alpha_n(x_n)\beta_{n+1}(x_{n+1}) p\left(x_{n+1}, y_{n+1} \left| x_n, y_n \right.\right),\\
p\left(x_{n+1} \left| x_n, \boldsymbol{y}_1^N \right.\right) &= \frac{p\left(x_n, x_{n+1} \left| \boldsymbol{y}_1^N \right.\right)}{p\left(x_n \left| \boldsymbol{y}_1^N \right.\right)}.
\end{aligned}$$

Table 1: Parameters characterizing precisely f_1 and f_2, which are of type-III and type-VI according to Pearson's system of distributions (see Appendix A). Means m_1^1 and m_2^1 are not specified here since experiments will be conducted according to variations of $\delta = |m_1^1 - m_2^1|$. Parameters b_0, b_1, ξ, λ, a_1, a_2, p_1 and p_2 are defined in Appendix A.

	m^1	m^2	β^1	β^2	b_0	b_1	ξ	λ
f_1	m_1^1	1.0	0.25	3.38	1.0	0.25	-9.5	6.37

	m^1	m^2	β^1	β^2	a_1	a_2	p_1	p_2	λ
f_2	m_2^1	1.0	1.00	4.70	-16.9	-2.34	-50.6	8.13	-0.95

5. GICE algorithm evaluation

This Section aims to evaluate particular GICE algorithms in the context of data classification with two classes ($\Omega = \{1, 2\}$). In a first series of experiments, we consider an HMM-DN and experiment the margin and copula recovering performances of GICE with respect to the distance between margins' mean. The experiment makes use of the Pearson's system of distributions, summarized in Appendix A. In a second series of experiments, we study in what situations the use of HMM-DN can improve the results obtained with classical HMM-INs. Finally, in last subsection, we provide a series of comparative results regarding the segmentation of a SAR image showing burn plots in Rondonia, Brazil.

5.1. HMM-DNs estimation and restoration

In this experimental setting, we consider fixed values for $p_{1,1} = p_{2,2} = 0.45$ and for $p_{1,2} = p_{2,1} = 0.05$, variations of them being considered in Section 5.2.

A two-classes HMM-DN is defined with 2 margins and 4 copulas. The margins f_1 and f_2 considered here are specified in Table 1, where m_i^1 denotes the mean, m_i^2 the variance, $\sqrt{\beta_i^1}$ the skewness and β_i^2 the kurtosis. Other parameters refer to the description of type-III and type-VI distributions according to Pearson's system (see Appendix A). The density of margins used in experiments are drawn in Fig. 1. The three copulas $c_{1,1}$, $c_{1,2} = c_{2,1}$ and $c_{2,2}$ involved in the

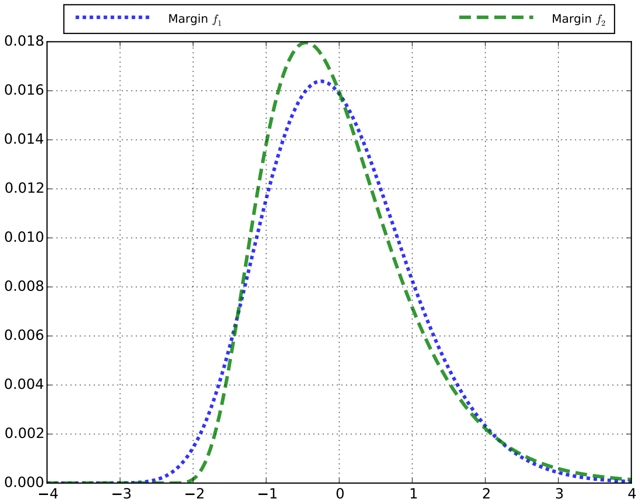


Figure 1: Density of margins used in experiments in Sections 5.1 and 5.2 (f_1: Gamma; f_2: second kind Beta). Parameters are specified in Table 1, with $m_1^1 = m_2^1 = 0$.

model were set to be respectively of Gumbel, Gaussian and Clayton types, with Kendall's tau given by $\tau_{1,1} = 0.1$, $\tau_{1,2} = \tau_{2,1} = 0.3$ and $\tau_{2,2} = 0.7$. Table 2 gives the details about the one-parameter families of copulas considered in this paper.

Data $\boldsymbol{X}_1^N = \boldsymbol{x}_1^N$ and $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$ were sampled as specified in Section 4.1, and the model was identified from $\boldsymbol{Y}_1^N = \boldsymbol{y}_1^N$ with GICE according to the algorithm described in Section 3. The sets of possible shapes for the margins was fixed to $\mathcal{F}_1 = \mathcal{F}_2 = \{\text{Gamma}, \text{Inverse Gamma}, \text{Second kind Beta}\}$. These three distributions correspond to a sub-set of Pearson's system of distributions (see remark below) so that, according to Example 3.2, the choice of the shapes and their parameters are performed simultaneously. More precisely, to find the margins at next iteration of GICE algorithm –points (c) and (d) of the GICE procedure in Section 3– one uses $\boldsymbol{y}^i(\boldsymbol{x}_1^N)$ (see 10) to classically estimate, for $i = 1, 2$, the four first moments $m_i^1 = E[Y_1|X_1 = i]$, and $m_i^s = E[(Y_1 - m_i^1)^s|X_1 = i]$ for $s = 2, 3, 4$, which gives β_i^1 and β_i^2.

According to the general theory [47], the knowledge of (β_i^1, β_i^2) gives the

Table 2: One-parameter copulas $c_p\left(y_1, y_2; \alpha\right)$ used in this work. "Arch" means Archimedean, and "14" in "Arch14" is the order of appearance in [10].

p	Name	pdf c_p	Kendall's τ
0	Product	$c_0(y_1, y_2) = 1$	0
1	Gauss	$c_1(y_1, y_2) = \frac{1}{\sqrt{1-\alpha^2}}\ \exp\left(-\frac{1}{2}\ \boldsymbol{\xi}^T\ (\boldsymbol{\rho} - \boldsymbol{I})\ \boldsymbol{\xi}\right)$ where $\xi_i = \phi^{-1}(y_i)$ with ϕ the standard normal distribution, $\boldsymbol{\rho} = \begin{pmatrix} 1 & \alpha \\ \alpha & 1 \end{pmatrix}$ and $\boldsymbol{I}$ are the 2×2 correlation and identity matrices.	$\frac{2}{\pi} \operatorname{asin} \alpha$
2	Gumbel	$c_2(y_1, y_2) = \frac{t_1}{y_1\ \ln(y_1)}\ \frac{t_2}{y_2\ \ln(y_2)}\ (\alpha - 1 + t_1 + t_2)^{\frac{1}{\alpha}}\ (t_1 + t_2)^{\frac{1}{\alpha}-2}\ \exp\left(-(t_1 + t_2)^{\frac{1}{\alpha}}\right)$ where $t_1 = (-\ln(y_1))^{\alpha}$ and $t_2 = (-\ln(y_2))^{\alpha}$.	$1 - \frac{1}{\alpha}$
3	Cubic Section	$c_3(y_1, y_2) = 1 + 2\alpha\ \Big((1-y_1)(1-y_2)(-8y_2y_1 + 2y_1 + 2y_2 + 1)$ $+y_1(1-y_2)(4y_2y_1 - y_1 - 2y_2 - 1) + (1-y_1)y_2(4y_2y_1 - 2y_1 - y_2 - 1)$ $+y_1y_2(-2y_2y_1 + y_1 + y_2 + 1)\Big)$	$\frac{2}{3}\alpha - \frac{2}{75}\alpha^2$
4	Clayton	$c_4(y_1, y_2) = (1+\alpha)\ y_1^{-1-\alpha}\ y_2^{-1-\alpha}\ \left(-1 + y_1^{-\alpha} + y_2^{-\alpha}\right)^{-\frac{1}{\alpha}-2}$	$\frac{\alpha}{\alpha+2}$
5	Arch14	$c_5(y_1, y_2) = t_1\ t_2\ (t_1 + t_2)^{\frac{1}{\alpha}-2}\ \left(1 + (t_1 + t_2)^{\frac{1}{\alpha}}\right)^{-2-\alpha}\ \frac{\left(\alpha - 1 + 2\alpha\,(t_1 + t_2)^{\frac{1}{\alpha}}\right)}{\alpha y_1 y_2\ \left(y_1^{\frac{1}{\alpha}} - 1\right)\ \left(y_2^{\frac{1}{\alpha}} - 1\right)}$ where $t_1 = \left(y_1^{-\frac{1}{\alpha}} - 1\right)^{\alpha}$ and $t_2 = \left(y_2^{-\frac{1}{\alpha}} - 1\right)^{\alpha}$	$\frac{2\alpha-1}{2\alpha+1}$

family which f_i belongs to (among eight families forming the Pearson's system of distributions), and the additional knowledge of m_i^1 characterizes f_i. More precisely, using the rules specified for each distribution in Appendix A, we first identify the distribution family of f_i from $\hat{\beta}_i^1$ and $\hat{\beta}_i^2$, and then, using $\hat{m}_i^1$ and formulas for each identified distribution, we precisely identify the shape of f_i.

Besides, we consider for all $i, j \in \Omega$ the same set of six possible shapes for copulas given by $\mathcal{G}_{1,1} = \mathcal{G}_{1,2} = \mathcal{G}_{2,1} = \mathcal{G}_{2,2} = \{$Product, Gaussian, Gumbel, Cubic section, Clayton, Arch14$\}$. All those copula families are detailed in Table 2. Except for the "Product" family, reduced to one element, a copula is entirely defined by its Kendall's tau τ, which can be classically estimated from concordance (c) and discordance (d) rates computed from $\boldsymbol{y}_{ij}(\boldsymbol{x}_{\boldsymbol{1}}^{\boldsymbol{N}})$:

$$\hat{\tau}_{i,j} = 2\frac{c-d}{n(n-1)},$$

where n is the sample size. Then, for each $\hat{\tau}_{i,j}$, we first compute the five possible $\hat{\alpha}_{i,j}$ for each of the five considered families by inverting the formulas in the last column in Table 2, and then apply the Pseudo Maximum Likelihood rule (11)

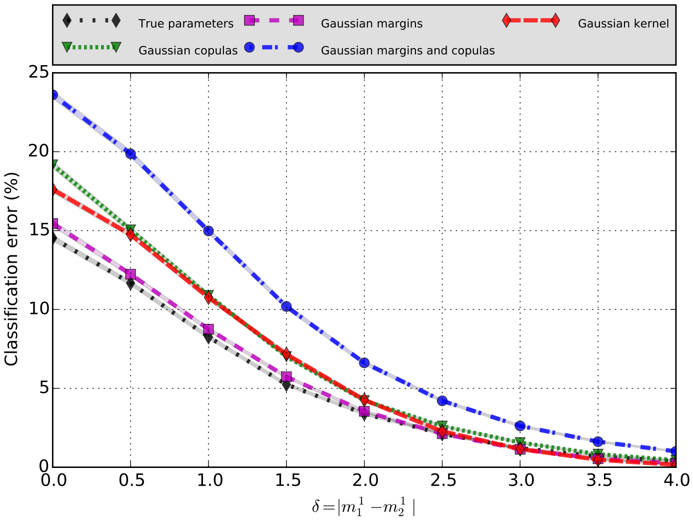


Figure 2: MPM classification error rates of five algorithms (true parameters) according to the gap δ between the two margins means (see text for details).

to select the best-fitting one.

The classification error rates presented hereafter are means of 100 independent experiments. An experiment consists in the simulation of $N = 3000$ data according to the HMC-DN model specified above, and its restoration according to different algorithms. Figures 2, 3 and 4 present the evolution of error rates of those algorithms according to the gap $\delta = |m_2^1 - m_2^1|$ between the means of the two margins involved (see Table 1). In each figure, the black plot (diamond marks) reports the error rate obtained with the true model (*i.e.* the restoration with the parameters used for simulation), which is thus a reference for all other methods.

Fig. 2 reports the error rates of four algorithms, assuming that the parameters are the true ones, for 'Gaussian margins', 'Gaussian copulas' and 'Gaussian margins and copulas', or are estimated from the ground-truth for 'Gaussian Kernel' (here GICE is not used):

1. 'Gaussian margins' (Magenta plot, square marks): the shape of the mar-

gins were assumed Gaussian, with means and variances given by m_1 and m_2 in Table 1, whereas the shape and the parameters of the copulas were the true ones;

2. 'Gaussian copulas' (Green plot, triangle marks): the shape of the 4 copulas were all assumed Gaussian, with the same Kendall's tau as the ones used for simulation ($\tau_{1,1} = 0.1$, $\tau_{1,2} = \tau_{2,1} = 0.3$ and $\tau_{2,2} = 0.7$), whereas the shape and the parameters of the margins were the true ones;
3. 'Gaussian margins and copulas'Blue plot (circle marks): Both margins and copulas were assumed Gaussian, with parameters set similarly to the two previous plots;
4. 'Gaussian kernel' (red plot, diamond marks): the shapes of the 4 class-conditional pdf $p\left(y_1, y_2 \left| x_1, x_2 \right.\right)$ were estimated using kernel density estimation. We used Gaussian kernels with no correlation and the $d = 2$ bandwidths were estimated using Scott's rule $\widehat{h_i} = n^{-1/(d+4)}\widehat{\sigma_i}$, with n the size of the sample and $\widehat{\sigma_i}$ the standard deviation for dimension i.

Regarding the first three plots, assuming the Gaussianity of either margins or copulas degrades the classification performances. Assuming full-Gaussianity when data are not Gaussian, which is often assumed in applications, leads to very poor results (blue plot): Gaussian approximations do not allow to capture the complexity and richness of the simulated data. At least in this experiment, when the copulas are the true ones, the margins shape has little influence on results (magenta plot). Finally, the Gaussian kernel plot (red) shows similar performances than the 'Gaussian copulas' configuration (green).

It is now interesting to measure the GICE performance for selecting shapes and estimating their parameters with respect to the classification error rates. To get the results reported in Fig. 3, the GICE was initialized from the parameters obtained with a Kmeans algorithm, and stopped after 100 iterations, assuming convergence. Fig. 3 reports the performance of three algorithms:

- 'GICE' (green plot, triangle marks): all shapes were automatically selected within the set of possible shapes by the GICE algorithm;

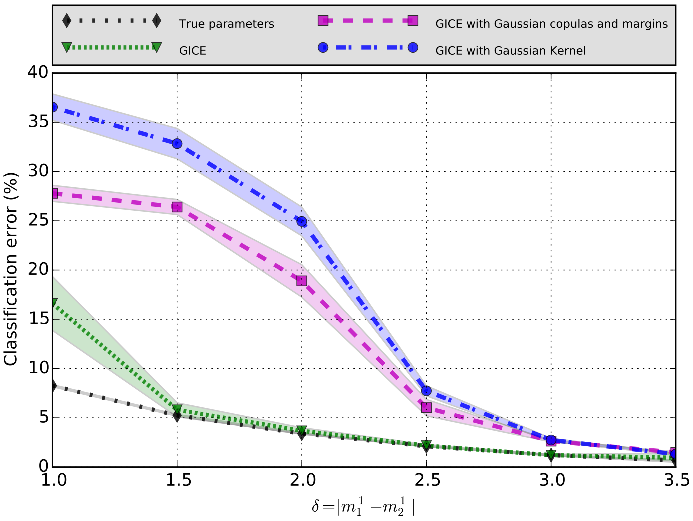


Figure 3: MPM classification error rate of five algorithms according to δ (see text for details). The results are means $\hat{\mu}$ of 100 experiments. The shaded envelop associated to each curve represents the 95% confidence interval of $\hat{\mu}$: $\hat{\mu} \pm 1.96 \frac{\hat{\sigma}}{\sqrt{100}}$

- ‘GICE with Gaussian copulas and margins’ (magenta plot, square marks): the shapes of the margins and copulas were all assumed Gaussian, ICE algorithm only performing parameters estimation;

- ‘GICE with Gaussian Kernel’ (blue plot, circle marks): the shapes of the 2D class-conditional densities were estimated by ICE using a simple Kernel density estimation algorithm (Gaussian Kernel, no correlation). The bandwidths were estimated using Scott’s rule at each ICE iteration.

The performance of the ‘GICE’ algorithm (green plot) is almost optimal since it is able to reach the performances of the reference, except when $\delta < 1.5$ in which case the mixture becomes too complex to be retrieved. Nevertheless, it gives very interesting results compared to the two other unsupervised algorithms (magenta and blue plots), allowing to divide the error rate up to 5 for $\delta = 1.5$. Hence, at least in this experiment, the automatic selection of the right copulas and the right margins are required to reach optimal performances. It is interest-

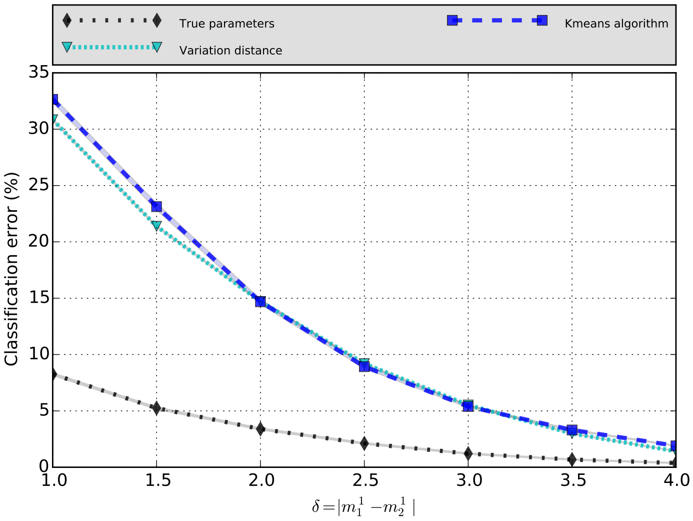


Figure 4: MPM classification error rate of two algorithms (‘Variation distance’ and ‘Kmeans algorithm’) according to δ (see text for details).

ing to note that the configurations represented by the magenta and blue plots give similar performances than the Kmeans and the “variation distance” classification algorithms, the last one being obtained without Markovianity, estimating each X_n from each Y_n by a suited ICE algorithm (simple mixture model), see Fig. 4.

As a conclusion for this experiment, we may state that the GICE algorithm we propose gives very satisfying results in HMM-DNs, even when the mixture to be restored is very difficult. This nice behaviour is confirmed by other similar experiments not reported in the paper. Let us note that the computational burden of the algorithm depends on the number of copula shapes which are evaluated at each iteration of the GICE algorithm. The selection based on PLM criterion (11) can be time-consuming, but the algorithm remains about ten times less time consuming that the kernel-based estimation method, whereas being much more performing. Otherwise, the selection of margins based on Pearson’ system is done at nearly no supplementary cost.

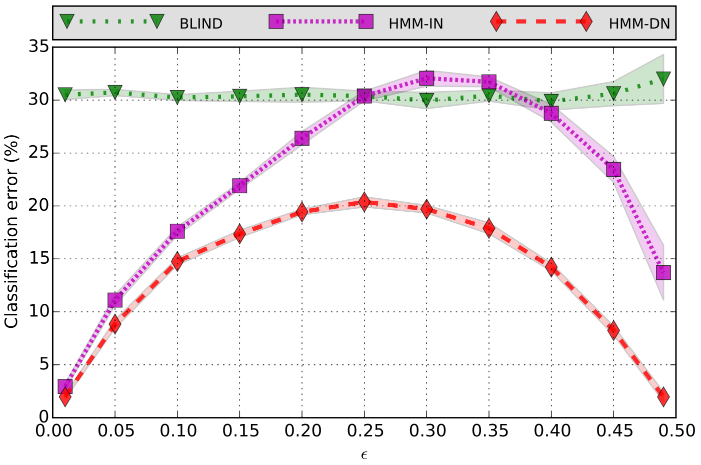


Figure 5: Classification error rate according to the joint a priori probability $\epsilon = p(x_1 = 1, x_2 = 1) = p(x_1 = 2, x_2 = 2)$ for 3 different models with decreasing generality: HMM-DN, HMM-IN, BLIND (means of 10 experiments of $N = 3000$ data).

5.2. Comparison with HMM-INs

The aim of this second experiment is to evaluate the interest of using HMM-DNs, which are not HMM-INs, *i.e.* in which $p(y_2 | x_2, y_1, x_1) \neq p(y_2 | x_2)$. We provide a study detailing the comparison *w.r.t.* $p(x_1, x_2)$ in a simple two-classes case. The parameters are here assumed to be known.

Let us still consider the HMM-DN specified by the four copulas and the two margins in previous sub-section. Parametrizing the joint a priori probability of $\boldsymbol{X}_1^N$ according to $p_{1,1} = p_{2,2} = \epsilon$ and $p_{1,2} = p_{2,1} = 1 - \epsilon$, we study the influence of ϵ value on the restoration of $N = 3000$ simulated data using the following algorithms, with decreasing modelling capabilities:

- HMM-DN (red plot, diamond marks): the HMM-DN model given by the four copulas, the two margins and matrix $p_{i,j}$;
- HMM-IN (magenta plot, square marks): the classic HMM model defined with margins f_1 and f_2 and matrix $p_{i,j}$;

- BLIND (green plot, triangle marks): the observations are assumed independent; the model is parametrized by margins f_1 and f_2 and the weights of the mixture (0.5 and 0.5).

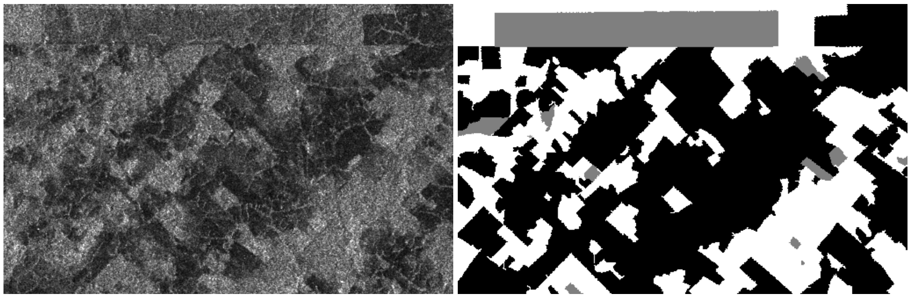
According to results reported in Fig. 5, we can state the following:

- Using HMM-DNs is always of interest, *i.e.* whatever the value of ϵ. The best gain is obtained for $\epsilon = 0.40$, the HMM-DN error being of 14% while the HMM-IN one is of 29%.
- The classic HMM-INs are quite inefficient, except for ϵ, *i.e.* inferior to 0.05. As HMM-INs are very simple, this could be of interest in such particular cases;
- The case $\epsilon = 0.25$ is of special interest as it is very different from the usual models. Indeed, the hidden variables $X_1, X_2, \dots, X_N$ are independent but, as observations $Y_1, Y_2, \dots, Y_N$ are dependent conditionally on $\boldsymbol{X}_1^N$ in HMM-DNs, $X_1, X_2, \dots, X_N$ are dependent conditionally on $\boldsymbol{Y}_1^N$ and thus the Markovianity of the couple $(\boldsymbol{X}_1^N, \boldsymbol{Y}_1^N)$ allows to improve blind and HMM-INs classifications. This is not the case in HMM-INs and we see here a particular aspect of the interest of HMM-DNs with respect to HMM-INs.

5.3. Unsupervised image segmentation

This section is intended to illustrate the use of automatic copulas and margins selection in HMC-DN for unsupervised image segmentation. We focus on the JERS1 Synthetic Aperture Radar (SAR) image of Rondonia, Brazil, in Fig. 6a. The image is a 3 looks amplitude image with 256×256 pixels, and $25m \times 25m$ soil resolution. SAR images are known to be very challenging due to the speckle that degrades the image with non-Gaussian noises. The image was manually segmented by an expert into 3 classes (burn plot, cultivation, and dense forest), *cf.* Fig. 6b.

To segment the image in 3 classes, we



(a) (b)

Figure 6: (a) 3-look JERS1 image. (b) Ground truth.

Table 3: Confusion matrices (in %) (a) for the HMC-DN model using GICE, and (b) for a fully-Gaussian HMD-DN model using classical ICE.

$$\begin{pmatrix} \mathbf{68.2} & 15.1 & 16.7 \\ 8.2 & \mathbf{77.1} & 14.6 \\ 8.1 & 10.0 & \mathbf{81.9} \end{pmatrix} \qquad \begin{pmatrix} \mathbf{41.9} & 42.6 & 15.5 \\ 4.1 & \mathbf{79.5} & 16.3 \\ 2.4 & 20.9 & \mathbf{76.7} \end{pmatrix}$$

(a) Overall error rate : 26.0% (b) Overall error rate : 41.9%

- apply the Hilbert-Peano scan [39] to get a 1D vector of data;
- apply the GICE algorithm and performed MPM-classification to get a 1D vector of class data, assuming
  - $\mathcal{F}_1 = \mathcal{F}_2 = \mathcal{F}_3 = \{\text{Gamma}, \text{Inverse Gamma}, \text{Second kind Beta}\}$ for margins, and
  - $\mathcal{G}_{1,1} = \ldots = \mathcal{G}_{3,3} = \{$Product, Gaussian, Gumbel, Cubic section, Clayton, Arch14$\}$ for copulas.
- transform the segmented 1D data into a 2D image using inverse scanning.

The result of segmentation with 3 classes is shown in Fig. 7a, with a confusion matrix reported in Table 3(a). The segmentation with 3 classes leads to

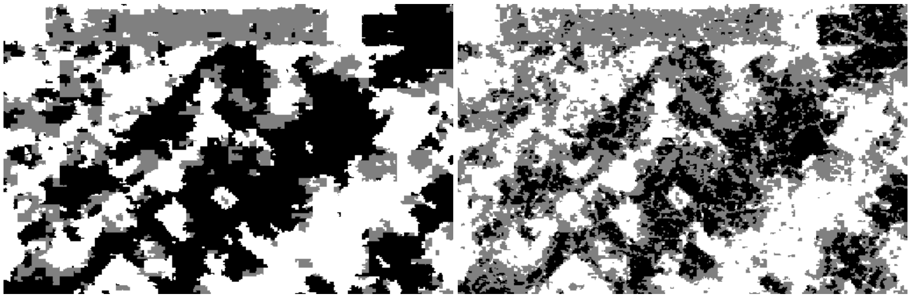


Figure 7: HMC-DN segmentation (a) within GICE framework with automatic margins and copulas selection, (b) within classical ICE framework with Gaussian margins and copulas.

- 9 different copulas $c_{1,1}, \ldots, c_{3,3}$, all of them being of Gumbel-type, and
- 3 margins f_1, f_2, and f_3, all of them being of Pearson's Type-VI .

The good performances of the algorithm can be compared *w.r.t* the segmentation obtained with a fully-Gaussian HMC-DN model with a parameters estimation performed using a classical ICE algorithm, see Fig. 7b and the corresponding confusion matrix reported in Table 3(b).

6. Conclusion

Classic Hidden Markov models are widely used in a number of situations. Considering dependent noise brings additional efficiency; however, it is not easy to handle with in non-Gaussian cases. Introducing copulas allows to consider large possibilities of different hidden Markov models. Extending works in [29, 38, 37, 39], we proposed here a general model's identification method from the only observed data. Experiments presented show the interest of copulas-based Markov models with respect to the classic ones, and the efficiency of the model's identification method proposed. In particular, at least in the experimental setting considered here, the automatic selection of both copulas and margins outperforms the results obtained using a Gaussian kernel representation for data-driven densities. It might be interesting to pursue the comparison with "Bayesian non parametric methods", such as Dirichlet Processes [50]. Nevertheless, in unsupervised context considered, it appears that Hidden Markov models with *dependent non-Gaussian* noise based methods clearly improve those based on the classic HMMs, as illustrated with a real SAR image.

In this paper, we considered mono-sensor cases and copulas were used at the temporal level. They may also be used, in hidden Markov context, at vectorial level, modelling dependencies among sensors at a given time [31, 33, 34, 35]. As perspective, one may consider to use copulas in hidden Markov models on both temporal and vectorial levels simultaneously. Another perspective is to consider extensions of the discrete hidden or pairwise Markov models considered to fuzzy ones, as introduced in [51, 52]. Finally, ICE has been successfully extended to long memory hidden Markov models [45], and thus considering copulas and GICE in such models would possibly be an interesting perspective .

References

[1] L. E. Baum, T. Petrie, G. Soules, N. Weiss, A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains, Ann. Math. Statist. 41 (1) (1970) 164–171.

[2] L. R. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proc. of the IEEE 77 (2) (1989) 257–286.

[3] O. Cappé, E. Moulines, T. Rydén, Inference in Hidden Markov Models, Springer-Verlag, 2005.

[4] Y. Ephraim, N. Merhav, Hidden Markov processes, IEEE Trans. on Inf. Theory 48 (6) (2002) 1518–1569.

[5] I. L. MacDonald, W. Zucchini, Hidden Markov and other models for discrete-valued time series, Monographs on statistics and applied probability, Chapman & Hall, London, New York, 1997.

[6] R. Bhar, S. Hamori, Hidden Markov models: applications to financial economics, Advanced Studies in Theoretical and Applied Econometrics Series, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2004.

[7] R. S. Mamon, R. J. Elliott, Hidden Markov models in finance, Springer, New York, 2007.

[8] T. Koski, Hidden Markov models for bioinformatics, Computational biology, Kluwer Academic Publishers Norwell, MA, Dordrecht, Teh Netherlands, 2001.

[9] M. Vidyasagar, Hidden Markov processes: theory and applications to biology, Princeton Series in Applied Mathematics, Princeton University Press, Princeton, NJ, 2014.

[10] R. Nelsen, An Introduction to Copulas, Springer, Springer Series in Statistics, Second edition, New York, USA, 2005.

[11] A. Sklar, Fonctions de répartition à n dimensions et leurs marges, Publications de l'Institut de Statistique de l'Université de Paris 8 (1959) 229–231.

[12] S. Demarta, A. J. McNeil, The t copula and related copulas, Int. Statistical Review 73 (2005) 111–129.

[13] C. Genest, J. Mackay, Copules archimédiennes et familles de lois bidimensionnelles dont les marges sont données, Canadian Journal of Statistics 14 (2) (1986) 145–159.

[14] A. K. Nikoloulopoulos, D. Karlis, Copula model evaluation based on parametric bootstrap, Comput. Stat. & Data Analysis 52 (7) (2008) 3342–3353.

[15] Y. Noh, K. Choi, I. Lee, Identification of marginal and joint CDFs using Bayesian method for RBDO, Structural and Multidisciplinary Optimization 40 (2010) 35–51.

[16] X. Chen, Y. Fan, Estimation of copula-based semiparametric time series models, Journal of Econometrics 130 (2) (2006) 307–335.

[17] O. C. da Silva Filho, F. A. Ziegelmann, M. J. Dueker, Modeling dependence dynamics through copulas with regime switching, Insurance: Mathematics and Economics 50 (3) (2012) 346–356.

[18] C. Genest, B. Rémillard, D. Beaudoin, Goodness-of-fit tests for copulas: A review and a power study, Insurance: Mathematics and Economics 44 (2009) 199–213.

[19] C. Genest, E. Masiello, K. Tribouley, Estimating copula densities through wavelets, Insurance: Mathematics and Economics 44 (2009) 170–181.

[20] E. Jondeau, M. Rockinger, The copula-GARCH model of conditional dependencies: An international stock market application, J. of Int. Money and Finance 25 (5) (2006) 827–853.

[21] Q. Xiaomei, Z. Jie, S. Xiaojing, Archimedean copula estimation and model selection via l1-norm symmetric distribution, Insurance: Mathematics and Economics 46 (2010) 406–414.

[22] S. Iyengar, P. Varshney, T. Damarla, A parametric copula-based framework for hypothesis testing using heterogeneous data, IEEE Trans. on Sig. Proc. 59 (5) (2011) 2308–2319.

[23] G. Mercier, G. Moser, S. Serpico, Conditional copula for change detection on heterogeneous SAR data, IEEE Trans. on Geoscience and Remote Sensing 46 (5) (2008) 1428–1441.

[24] Y. Stitou, N. Lasmar, Y. Berthoumieu, Copulas based multivariate gamma modeling for texture classification, in: IEEE Int. Conf on Acoustics, Speech and Sig. Proc. (ICASSP'09), 2009, pp. 1045–1048.

[25] A. Sundaresan, P. Varshney, Location estimation of a random signal source based on correlated sensor observations, IEEE Trans. on Sig. Proc. 59 (2) (2011) 787–799.

[26] M. A. Ben Alaya, F. Chebana, T. B. M. J. Ouarda, Probabilistic Gaussian copula regression model for multisite and multivariable downscaling, J. of Climate 27 (9) (2014) 3331–3347.

[27] A.-C. Favre, S. El Adlouni, L. Perreault, N. Thiémonge, B. Bobée, Multivariate hydrological frequency analysis using copulas, Water Resources Research 40.

[28] S. Grimaldi, F. Serinaldi, Asymmetric copula in multivariate flood frequency analysis, Advances in Water Resources 29 (2006) 1155–1167.

[29] N. Brunel, W. Pieczynski, Unsupervised signal restoration using hidden Markov chains with copulas, Sig. Proc. 85 (12) (2005) 2304–2315.

[30] N. Brunel, J. Lapuyade-Lahorgue, W. Pieczynski, Modeling and unsupervised classification of multivariate hidden Markov chains with copulas, IEEE Trans. on Automatic Control 55 (2) (2010) 338–349.

[31] F. Flitti, C. Collet, A. Joannic-Chardin, Unsupervised multiband image segmentation using hidden markov quadtree and copulas, in: IEEE Int. Conf. on Image Processing (ICIP'05), Genova, Italy, 2005, pp. 634–637.

[32] W. Wang, O. Okhrin, W. K. Hrdle, Hidden Markov structures for dynamic copulae, forthcoming in Econometric Theory (2014).

[33] F. Flitti, C. Collet, E. Slezak, Image fusion based on pyramidal multiband multiresolution markovian analysis, Signal, Image and Video Processing 3 (3) (2009) 275–289.

[34] A. Voisin, V. Krylov, G. Moser, S. B. Serpico, J. Zerubia, Classification of very high resolution SAR images of urban areas using copulas and texture in a hierarchical Markov random field model, IEEE Geoscience and Remote Sensing Letters 10 (1) (2013) 96–100.

[35] A. Voisin, V. Krylov, G. Moser, S. B. Serpico, J. Zerubia, Supervised classification of multi-sensor and multi-resolution remote sensing images with

a hierarchical copula-based approach, IEEE Trans. on Geosci. and Rem. Sens. 52 (6) (2014) 3346–3358.

[36] A. Ekin, A. M. Tekalp, Automatic soccer video analysis and summarization, IEEE Trans. on Image Processing 12 (2003) 796–807.

[37] S. Derrode, W. Pieczynski, Unsupervised data classification using pairwise Markov chains with automatic copulas selection, Comput. Stat. & Data Analysis 63 (2013) 81–98.

[38] Y. Delignon, A. Marzouki, W. Pieczynski, Estimation of generalized mixture and its application in image segmentation, IEEE Trans. on Image Processing 6 (10) (1997) 1364–1375.

[39] N. Giordana, W. Pieczynski, Estimation of generalized multisensor hidden Markov chains and unsupervised image segmentation, IEEE Trans. on Pattern Analysis and Machine Intelligence 19 (5) (1997) 465–475.

[40] S. Derrode, W. Pieczynski, Signal and image segmentation using pairwise Markov chain, IEEE Trans. on Sig. Proc. 52 (9) (2004) 2477–2489.

[41] W. Pieczynski, Pairwise Markov chains, IEEE Trans. on Pattern Analysis and Machine Intelligence 25 (2003) 634–639.

[42] P. Lanchantin, J. Lapuyade-Lahorgue, W. Pieczynski, Unsupervised segmentation of randomly switching data hidden with non-Gaussian correlated noise, Sig. Proc. 91 (2) (2011) 163–175.

[43] J. Lapuyade-Lahorgue, W. Pieczynski, Unsupervised segmentation of hidden semi-Markov non stationary chains, Sig. Proc. 92 (1) (2012) 29–42.

[44] Y. Ephraim, B. L. Mark, Bivariate markov processes and their estimation, Foundations and Trends in Sig. Proc. 6 (1) (2013) 1–95.

[45] P. Lanchantin, J. Lapuyade-Lahorgue, W. Pieczynski, Unsupervised segmentation of triplet Markov chains hidden with long-memory noise, Sig. Proc. 88 (5) (2008) 1134–1151.

[46] J. Delmas, An equivalence of the EM and ICE algorithms for exponential family, IEEE Trans. on Sig. Proc. 45 (10) (1997) 2613–2615.

[47] N. Johnson, S. Kotz, Continuous univariate distribution, Tome I and II, Wiley-interscience, second edition, 1994.

[48] D. Huard, G. Évin, A. Favre, Bayesian copula selection, Comput. Stat. & Data Analysis 51 (2006) 809–822.

[49] L. Devroye, Non-Uniform Random Variate Generation, Springer-Verlag, New York, New York, USA, 1986.

[50] F. Caron, M. Davy, A. Doucet, E. Duflos, P. Vanheeghe, Bayesian inference for linear dynamic models with Dirichlet process mixtures, IEEE Trans. on Signal Processing 56 (1) (2008) 71–84.

[51] C. Carincotte, S. Derrode, S. Bourennane, Unsupervised change detection on SAR images using fuzzy hidden Markov chains, IEEE Trans. on Geoscience and Remote Sensing 44 (2) (2006) 432–441.

[52] F. Salzenstein, C. Collet, S. Le Cam, M. Hatt, Non stationnary fuzzy Markov chain, Pattern Recognition Letters 28 (16) (2007) 2201–2208.

Appendix A. Brief recall on Pearson' system of distributions

A pdf f on $\mathbb{R}$ belongs to Pearson's system if it satisfies

$$\frac{\partial}{\partial x}\ln f(x) = \frac{(x-\lambda)+a}{b_2(x-\lambda)^2+b_1(x-\lambda)+b_0}, \tag{A.1}$$

with

$$\begin{aligned} b_0 &= \frac{4\beta^2-3\beta^1}{10\beta^2-12\beta^1-18}m^2, \quad b_2 = \frac{2\beta^2-3\beta^1-6}{10\beta^2-12\beta^1-18} \\ b_1 &= \sqrt{m^2\beta^1}\frac{\beta^2+3}{10\beta^2-12\beta^1-18} = a, \end{aligned}$$

and m^i denotes the moment of order i, $\sqrt{\beta^1}$ the skewness, and β^2 the kurtosis:

$$\beta_i^1 = \frac{(m_i^3)^2}{(m_i^2)^3}, \quad \beta_i^2 = \frac{m_i^4}{(m_i^2)^2}. \tag{A.2}$$

The variation of the parameters a, λ, b_0, b_1 and b_2 provides distributions of eight different shapes and, for each shape, defines the parameters fixing a given distribution. The pdf used in experiments constitute a subset of Pearson's system made of 3 densities that are detailed in the following.

Pearson type-III distribution. A distribution is said to be of type-III if $\beta^2 = \frac{3}{2}(\beta^1 + 2)$. Hence, denoting,

$$\xi = \frac{3(\beta^2 - 1)}{2(3 - \beta^2)}, \quad \lambda = \frac{b_0}{b_1} - \xi b_1,$$

the random variate $b_0 + b_1(x - \lambda - m^1)$ is Gamma(ξ, θ)-distributed with $\theta = b_1^2$.

Pearson type-V distribution. A distribution is of type-V if $\beta^2 = -3\frac{2(\beta^1+4)^{1.5}+13\beta^1+16}{\beta^1-32}$. Hence, denoting,

$$c_1 = \frac{b_1}{2b_2}, \quad \lambda = -\frac{a - c_1}{1 - 2b_2},$$

the random variate $x - \lambda - m^1$ is distributed according to an inverse Gamma distribution IG(α, β) with parameters $\alpha = \frac{1}{b_2} - 1$ and $\beta = \frac{a-c_1}{b_2}$.

Pearson type-VI distribution. A distribution is said to be of type-V if values of β^1 and β^2 belong to the restriction of Pearson (β^1, β^2)-plane delimited by type-III and type-V distributions. Hence, denoting,

$$\begin{aligned}
\Delta &= b_1^2 - 4b_2b_0, \quad \lambda = (a_2 - a_1)\frac{p_2 + 1}{p_2 + p_1 + 2} - a_2, \\
a_1 &= \frac{-b_1 - \sqrt{\Delta}}{2b_2}, \quad a_2 = \frac{-b_1 + \sqrt{\Delta}}{2b_2}, \\
p_1 &= \frac{b_1 - a_1}{b_2(a_1 - a_2)}, \quad p_2 = \frac{b_1 - a_2}{b_2(a_2 - a_1)},
\end{aligned}$$

the random variate $\frac{x-\lambda-p^1-a_2}{a_2-a_1}$ is distributed according to a Beta prime distribution B'(α, β) with parameter $\alpha = p_2 + 1$ and $\beta = -p_2 - p_1 - 1$.