INSTITUT DE STATISTIQUE

BIOSTATISTIQUE ET

SCIENCES ACTUARIELLES

(ISBA)

UNIVERSITÉ CATHOLIQUE DE LOUVAIN

DISCUSSION
PAPER

2013/09

Nonparametric estimation of the tree structure of a nested Archimedean copula

SEGERS J. and N. UYTTENDAELE

Nonparametric estimation of the tree structure of a nested Archimedean copula

Johan Segers[a,1], Nathan Uyttendaele[a,1,*]

[a] *Université catholique de Louvain, Institut de Statistique, Biostatistique et Sciences Actuarielles, Voie du Roman Pays 20, B-1348 Louvain-la-Neuve, Belgium*

Abstract

One of the features inherent in nested Archimedean copulas, also called hierarchical Archimedean copulas, is their rooted tree structure. In this paper, a nonparametric, rank-based method to estimate this structure is developed. Our approach consists in representing the rooted tree structure as a set of trivariate structures that can be estimated individually. Indeed, for any triple of variables there are only four possible rooted tree structures and, based on a sample, a choice can be made by performing comparisons between the three bivariate margins of the empirical distribution of the triple. The set of estimated trivariate structures can then be used to build an estimate of the global rooted tree structure. This approach has the advantage that no assumptions about the nested Archimedean copula is required prior to the estimation of its structure.

Keywords: Archimedean copula, dependence, nested Archimedean copula, hierarchical Archimedean copula, rooted tree, Kendall distribution, nonparametric inference

1. Introduction

Archimedean copulas have become a popular tool for modeling or simulating bivariate data. However, because of their highly symmetric nature, they usually fail to properly model data in higher dimensions. Nested Archimedean copulas (NACs), or hierarchical Archimedean copulas, are an interesting attempt to overcome this drawback. They were first introduced by Joe (1997) and have been studied many times since, see for instance McNeil (2008), Hofert (2010),

[*] Corresponding author

Email addresses: `johan.segers@uclouvain.be` (Johan Segers), `nathan.uyttendaele@uclouvain.be` (Nathan Uyttendaele)

[1] This research is supported by contract "Projet d'Actions de Recherche Concertées" No. 12/17-045 of the "Communauté française de Belgique" and by IAP research network grant nr. P7/06 of the Belgian government (Belgian Science Policy).

Hering, Hofert, Mai and Scherer (2010), Hofert and Maechler (2011), Hofert and Pham (2012) or Okhrin, Okhrin and Schmid (2013).

The hierarchy of variables in a nested Archimedean copula is described through a rooted tree. Most often, the tree is taken as given from the context, for instance in Hofert (2010) or in Puzanova (2011). Okhrin, Okhrin and Schmid (2013) were the first to address the issue of reconstructing the tree from a sample, offering a parametric answer to the problem. In contrast, the method we propose is completely nonparametric and does not require the user to make any assumption on the NAC from which the tree structure must be estimated.

Sections 2 and 3 of this paper introduce Archimedean copulas and a nested Archimedean copulas. The fourth section adds a condition on nested Archimedean copulas to ensure the tree structure is always well identified.

Section 5 shows a convenient way to store a NAC tree structure which will be used throughout the rest of the paper, while Section 6 introduces a key point, namely that a NAC structure can always be represented as a set of trivariate NAC structures. That is, for a random vector of continuous random variables $(X_1, \ldots, X_d)$ with a NAC as joint copula, it is possible to get the tree structure of this nested Archimedean copula provided the tree structure of the nested Archimedean copula associated with each triple of variables (X_i, X_j, X_k) with distinct $i, j, k \in \{1, \ldots d\}$ is known.

Next, it is shown in Section 7 how the NAC structure of a triple of variables (X_i, X_j, X_k) can be estimated nonparametrically. The idea is to estimate the Kendall distribution associated with each pair of variables within the triple, these estimates allowing us to decide if each pair of variables has actually the same underlying Kendall distribution or not. If yes, then the tree structure of the triple is the trivial trivariate structure. If not, determining which pair has a different underlying Kendall distribution allows to assign the correct NAC tree structure to the triple of variables.

When the NAC tree structure of each of the $\binom{d}{3}$ triples has been estimated, it may happen that a proper d-variate NAC structure cannot be retrieved. How to deal with this issue is described in Section 8.

The performance of our approach is then assessed by means of a simulation study involving target structures in several dimensions. As part of this simulation study, some comparisons with the work of Okhrin, Okhrin and Schmid (2013) are also performed using their R package **HAC** (Okhrin and Ristig, 2012a).

Finally, an application section shows the usefulness of our method when applied on some financial data, and some remaining challenges are outlined in a discussion section.

2. Archimedean copulas

Let $(X_1, \ldots, X_d)$ be a vector of continuous random variables. The unique joint copula of this vector is defined as

$$C(u_1, \ldots, u_d) = P(U_1 \leq u_1, \ldots, U_d \leq u_d)$$

where $(U_1, \ldots, U_d) = (F_{X_1}(X_1), \ldots, F_{X_d}(X_d))$ and where $F_{X_1}, \ldots, F_{X_d}$ are the marginal cumulative distribution functions, or CDFs in short, of $X_1, \ldots, X_d$ respectively.

Archimedean copulas (ACs) are a class of copulas that admit the representation

$$C(u_1, \ldots, u_d) = \psi(\psi^{-1}(u_1) + \cdots + \psi^{-1}(u_d))$$

where ψ is called the generator and ψ^{-1} is its generalized inverse, with $\psi : [0, \infty) \to [1, 0]$; $\psi(0) = 1$ and $\psi(\infty) = 0$. In order for C to be a copula, the generator is required to be d-monotone on $[0, \infty)$ (McNeil and Nešlehová, 2009):

- $(-1)^k \psi^{(k)}(x) \geq 0$ for all $x \geq 0$; $k = 0, 1, \ldots, d-2$ and where $\psi^{(k)}$ is the kth derivative of $\psi(x)$;
- $(-1)^{d-2} \psi^{(d-2)}(x)$ is nonincreasing and convex.

The generators in the following table are among the most popular ones. All of them are completely monotone, that is, d-monotone for all integer $d \geq 2$.

Table 1: Some popular generators for Archimedean copulas.

name	generator $\psi(x)$	θ	τ
AMH	$(1-\theta)/(e^x - \theta)$	$\theta \in [0, 1)$	$1 - 2\left(\theta + (1-\theta)^2 \log(1-\theta)\right)/(3\theta^2)$
Clayton	$(1+x)^{-1/\theta}$	$\theta \in (0, \infty)$	$\theta/(\theta+2)$
Frank	$-\log(1 - (1 - e^{-\theta})e^{-x})/\theta$	$\theta \in (0, \infty)$	$\theta/(\theta+2)$
Gumbel	$\exp(-x^{1/\theta})$	$\theta \in [1, \infty)$	$(\theta-1)/\theta$
Joe	$1 - (1 - e^{-x})^{1/\theta}$	$\theta \in [1, \infty)$	$1 - 4\sum_{k=1}^{\infty} 1/(k(\theta k + 2)(\theta(k-1) + 2))$

The parameter θ in Table 1 allows to control the strength of dependence between any two variables of the related Archimedean copula. This is best understood by expressing Kendall's τ coefficient between any two variables of the related Archimedean copula in terms of θ (Hofert and Maechler, 2011), as done in the last column of the above table.

All margins of the same dimension of an AC are equal. This is because $C(u_1, \ldots, u_d)$ is a symmetric function in its arguments in the case of Archimedean copulas. For modeling purposes, this becomes an increasingly strong assumption as the dimension grows.

3. Nested Archimedean copulas

Asymmetries, allowing for more realistic dependencies, are obtained by plugging in Archimedean copulas into each other (Joe, 1997). For instance, in the two-dimensional Archimedean copula

$$C_{D_0}(u_1, \bullet) = \psi_{D_0}(\psi_{D_0}^{-1}(u_1) + \psi_{D_0}^{-1}(\bullet)),$$

the argument $\bullet$ can be replaced by another Archimedean copula, such as

$$\boldsymbol{C_{D_{23}}(u_2, u_3) = \psi_{D_{23}}(\psi_{D_{23}}^{-1}(u_2) + \psi_{D_{23}}^{-1}(u_3))}$$

in order to get a copula of the form

$$C_{D_0}(u_1, \boldsymbol{C_{D_{23}}(u_2, u_3)}) = \psi_0\big(\psi_{D_0}^{-1}(u_1) + \psi_{D_0}^{-1}(\boldsymbol{\psi_{D_{23}}(\psi_{D_{23}}^{-1}(u_2) + \psi_{D_{23}}^{-1}(u_3))})\big). \tag{3.1}$$

This last equation describes a copula where the marginal bivariate distribution of (U_2, U_3) is not the same as the marginal bivariate distribution of (U_1, U_2) or (U_1, U_3), provided the generators ψ_{D_0} and $\psi_{D_{23}}$ are different. If the joint CDF of (U_1, U_2, U_3) was a simple Archimedean copula, all the marginal bivariate distributions would have been the same. This allows to appreciate how the symmetry inherent in Archimedean copulas can be broken, although some leftover symmetry always remains, as the marginal bivariate distributions of (U_1, U_2) and (U_1, U_3) are the same.

The way Archimedean copulas are nested corresponds to a rooted tree structure, which will be referred to as the *NAC tree structure* or sometimes simply as the *structure* later. Nested Archimedean copulas, such as the one in (3.1), are defined through that rooted tree structure and through a collection of generators, one for each branching node in the tree. If the only nodes in the tree are the root and the leaves, then the copula is an Archimedean one, that is, a nested Archimedean copula with trivial structure and only one generator.

DEFINITION 3.1. *Let D_0 be a nonempty, finite set with $|D_0| = d$ elements. For concreteness, let $D_0 = \{U_1, \ldots, U_d\}$. Formally, a* rooted tree structure *λ on D_0 is a collection of nonempty subsets of D_0 such that*

(i) $D_0 \in \lambda$;

(ii) $\{a\} \in \lambda$ *for every* $a \in D_0$;

(iii) *if* $A, B \in \lambda$, *then either* $A \subset B$, $B \subset A$, *or* $A \cap B = \varnothing$.

The elements of λ are called the nodes *of the structure. The element D_0 of λ is called the* root node, *or* root *in short; the singleton elements $\{a\}$ of λ are called the* leaves. *The nodes of λ that are not leaves are called the* branching nodes. *If $A, B \in \lambda$ are such that $A \subset B$, $A \neq B$, and there is no $C \in \lambda$ such that $A \subset C \subset B$ and $C \neq A$ and $C \neq B$, then A is called a* child *of B and conversely B is called the parent of A. The set of children of B in λ is denoted by $\mathcal{C}(B, \lambda)$.*

For instance, the structure λ implied by Equation (3.1) is

$$\big\{\{U_1, U_2, U_3\}, \{U_2, U_3\}, \{U_1\}, \{U_2\}, \{U_3\}\big\}$$

and it can be graphically represented as shown in the picture below, where D_{23} is a convenient label for the subset $\{U_2, U_3\}$.

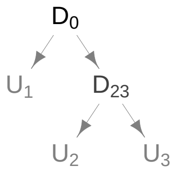

Figure 1: The tree structure implied by Equation (3.1). To ease the notation, the singletons $\{U_1\}$, $\{U_2\}$ and $\{U_3\}$ are denoted by U_1, U_2 and U_3.

In this structure, $\{U_2\}$ and $\{U_3\}$ are the children of D_{23} while $\{U_1\}$ and D_{23} are the children of D_0, the root node.

Let λ be a rooted tree on $D_0 = \{U_1, \ldots, U_d\}$. Suppose that for each $B \in \lambda$ with $|B| \geqslant 2$ we are given an Archimedean generator ψ_B, that is, we are given a generator for each branching node in the structure.

Define then recursively the function $C_B : [0,1]^{|B|} \to [0,1]$, with $B \in \lambda$, $|B| \geqslant 1$, by

$$C_B(u_b : b \in B) = \begin{cases} u_b & \text{if } B = \{b\} \\ \psi_B\Big(\sum_{A \in \mathcal{C}(B,\lambda)} \psi_B^{-1}\big(C_A(u_a : a \in A)\big)\Big) & \text{if } |B| \geqslant 2 \end{cases} \tag{3.2}$$

Definition 3.2. *A d-variate copula C_{D_0} is a* nested Archimedean copula (NAC) *if it is of the form C_B in* (3.2)*, with $B = D_0$.*

For any $A \subset D_0$ with $|A| \geqslant 2$, the copula C_A on the variables $(u_a : a \in A)$ turns out to be a nested Archimedean copula too. To describe its structure and its generators, we need a few more definitions.

Let λ be a NAC structure on D_0 and let T be a nonempty subset of D_0. The set T need not be a node of λ. The NAC structure λ induces a NAC structure on T by the following operation:

$$\lambda \sqcap T = \{T \cap B : B \in \lambda\} \setminus \{\varnothing\}.$$

That is, $\lambda \sqcap T$ is obtained by intersecting every node B of λ with T. Some of these intersections will be empty, and they are removed. Different nodes B_1 and B_2 of λ may have identical intersections $B_1 \cap T$ and $B_2 \cap T$ with T; since $\lambda \sqcap T$ is the collection of all intersections, identical intersections are counted only once.

It is easy to verify that this construction produces a tree structure on T: verification of (i), (ii), and (iii) in Definition 3.1 is immediate.

Let now T be a subset of D_0 containing at least two elements, that is $|T| \geq 2$. T does not need to be a node of λ. The *smallest common ancestor* (sca) of the elements of T is given by the intersection of all the nodes B in λ that contain T, that is,

$$\operatorname{sca}(T,\lambda) = \bigcap_{B\in\lambda : T\subseteq B} B$$

and it provides the smallest branching node through which the elements of T are linked up. For instance, looking back at Figure 1, one can see that the smallest common ancestor between U_2 and U_3 is D_{23}, while $\operatorname{sca}(\{U_1, U_2\}, \lambda) = D_0$ and $\operatorname{sca}(\{U_1, U_3\}, \lambda) = D_0$.

Let C_{D_0} be a d-variate nested Archimedean copula and let A be a nonempty subset of D_0, not necessarily a node in the tree λ. The marginal copula C_A on the variables in A is a nested Archimedean copula too. Its NAC structure is given by $\lambda \sqcap A$, and the generator function associated to a branching node T in $\lambda \sqcap A$ is given by $\psi_{\operatorname{sca}(T,\lambda)}$.

As appealing as it is, Definition 3.2 is unfortunately not sufficient to guarantee that C_{D_0} and its margins are copulas. A sufficient but not necessary condition was developed by Joe (1997) and McNeil (2008): the derivatives of $\psi_I^{-1} \circ \psi_J$ are required to be completely monotone for every pair of branching nodes I and J in the NAC structure such that J is a child of I. As an example, a sufficient condition for C_{D_0} in Equation (3.1) to be a proper copula is that the derivatives of $\psi_{D_0}^{-1} \circ \psi_{D_{23}}$ are completely monotone. Although this sufficient nesting condition was originally formulated only in the context of fully nested Archimedean copula structures, that is structures where each branching node has either two leaves as children, or one leave and another branching node, we assume this sufficient nesting condition to hold for any NAC structure.

The sufficient nesting condition is often easily verified if all generators appearing in the nested structure come from the same parametric family. For each family of Table 1, two generators ψ_I and ψ_J of the same family with corresponding parameters θ_I and θ_J will fulfill the sufficient nesting condition if $\theta_I \leq \theta_J$, assuming J is the child of I. Verifying the sufficient nesting condition if ψ_I and ψ_J do not belong to the same Archimedean family is usually harder, see for instance Hofert (2010).

4. Identifiability

Recall that a parameter θ (possibly infinite-dimensional) in a statistical model $(P_\theta : \theta \in \Theta)$, with P_θ a probability measure on a fixed space, is identifiable if $\theta_1 \neq \theta_2$ implies that $P_{\theta_1} \neq P_{\theta_2}$, that is, different parameters yield different distributions of the observable. For d-variate nested Archimedean copulas, the parameter θ consists of the pair

$$\big(\lambda, \{\psi_B : B \in \lambda, |B| \geqslant 2\}\big).$$

In this parametrization, the parameter θ is not identifiable, since replacing a generator function $\psi_B(x)$ by the function $\psi_B(ax)$, with $0 < a < \infty$, yields the same copula; that is, the generator functions are identifiable up to scaling only. This issue can be solved easily in different ways, for instance by requiring that $\psi_B(1) = 1/2$.

However, a more fundamental identifiability issue arises if some generator functions are not different. Consider for instance the tree λ implied by Equation (3.1), shown in Figure 1. If the generators ψ_{D_0} and $\psi_{D_{23}}$ are the same, say $\boldsymbol{\psi}$, then the nested Archimedean copula with parameter $(\lambda; \psi_{D_0}, \psi_{D_{23}})$ is

$$\begin{aligned} C_{D_0}(u_1, C_{D_{23}}(u_2, u_3)) &= \psi_0\big(\psi_{D_0}^{-1}(u_1) + \psi_{D_0}^{-1}(\psi_{D_{23}}(\psi_{D_{23}}^{-1}(u_2) + \psi_{D_{23}}^{-1}(u_3)))\big) \\ &= \boldsymbol{\psi}\big(\boldsymbol{\psi}^{-1}(u_1) + \boldsymbol{\psi}^{-1}(u_2) + \boldsymbol{\psi}^{-1}(u_3)\big) \end{aligned}$$

and actually describes an Archimedean copula with generator $\boldsymbol{\psi}$, that is, a nested Archimedean copula with trivial tree structure and single generator $\boldsymbol{\psi}$.

To avoid such identifiability issue we must require that for any two nodes A and B such that $A \subset B$ and $A \neq B$, meaning A is a descendant of B or conversely B is an ancestor of A, the bivariate Archimedean copulas generated by the generator functions ψ_A and ψ_B are different. If this holds, then the structure λ and the generators $\{\psi_B : B \in \lambda, |B| \geqslant 2\}$ can be identified (up to scaling) from a nested Archimedean copula C_{D_0}.

Note that some generator functions can still be identical. Consider for instance the structure in Figure 2:

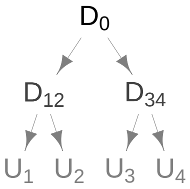

Figure 2: D_{12} is a convenient label for $\{U_1, U_2\}$, as well as D_{34} for the subset $\{U_3, U_4\}$. Again, we ease the notation by writing $U_1, ..., U_d$ instead of $\{U_1\}, ..., \{U_d\}$ for the singletons.

The generators associated to the nodes D_{12} and D_{34} can be identical, without simplification of the tree being possible.

Also note the implication of this identifiability condition on the sufficient nesting condition if all generators appearing in the nested structure come from the same parametric family. For each family of Table 1, two generators ψ_I and ψ_J of the same family with corresponding parameters θ_I and θ_J will fulfill the sufficient nesting condition *and* the identifiability condition if θ_I is *stricly less* than θ_J, assuming J is a child of I.

5. The smallest common ancestor matrix

Let λ be a NAC structure on D_0. Let $A, B \in \lambda$ be two distinct nodes. Recall that if $A \subset B$ and there is no $C \in \lambda$ such that $A \subset C \subset B$ and $C \neq A$ and $C \neq B$, then A is called a *child* of B and conversely B is called *the parent* of A.

The set of all children of a branching node B forms a partition of B, that is, taking the union of all children of a branching node B allows to reconstruct that branching node. As a consequence, every branching node has at least two children.

Also recall that if T is a subset of D_0 containing at least two elements, then the *smallest common ancestor* (sca) of the elements of T is given by the intersection of all the nodes B in λ that contain T, that is,

$$\operatorname{sca}(T, \lambda) = \bigcap_{B \in \lambda : T \subseteq B} B$$

and it provides the smallest branching node through which the elements of T are linked up.

Since the children of a branching node B form a partition of B and since each branching node has at least two children, it follows that each branching node can be reconstructed from the pairs of which it is the smallest common ancestor, that is, for every branching node B, we have

$$B = \bigcup \bigl\{ \{U_i, U_j\} \subset D_0 : U_i \neq U_j,\ \operatorname{sca}(\{U_i, U_j\}, \lambda) = B \bigr\}. \tag{5.1}$$

The relation "...has the same smallest common ancestor as ..." is an equivalence relation on the set of pairs $\{U_i, U_j\}$ of D_0. This relation induces a partition of the set of pairs into equivalence classes: two pairs $\{U_i, U_j\}$ and $\{U_k, U_l\}$ belong to the same equivalence class if and only if they are related, that is, if and only if they have the same smallest common ancestor in λ.

By Equation (5.1), the NAC structure λ can be reconstructed from the equivalence relation it induces on the set of pairs: every equivalence class of pairs corresponds to a branching node, the branching node being given by the union of the pairs in that equivalence class. Put differently, the union of all pairs within an equivalence class yields the branching node that is the smallest common ancestor for each pair in that equivalence class. Hence, every NAC structure λ on D_0 can be represented as a partition on the set of pairs of D_0 and the structure can be recovered from that partition.

A convenient way to display the equivalence classes is by the *smallest common ancestor matrix* or in short *sca matrix*: it is a d-by-d symmetric matrix containing the elements of D_0 in the rows and columns and whose element (i, j), with $i \neq j$, is the label of the node, in λ, which is the smallest common ancestor associated with the equivalence class to which the pair $\{U_i, U_j\}$ belongs. Put more simply: the element (i, j), with $i \neq j$, of the sca matrix is the name of the node in λ which is the smallest common ancestor of the pair $\{U_i, U_j\}$. Note that although the labels (the names) of the nodes of a given structure λ are arbitrary as long as they are different, we will always give the name D_0 to the root node in this paper, while a label such as D_{2379} means the related node is $\{U_2, U_3, U_7, U_9\}$. The diagonal of the smallest common ancestor matrix always remains empty.

As an example, the sca matrices for Figure 1 and for a trivariate Archimedean copula are given hereafter:

(i,j)	U_1	U_2	U_3
U_1		D_0	D_0
U_2	D_0		D_{23}
U_3	D_0	D_{23}	

(i,j)	U_1	U_2	U_3
U_1		D_0	D_0
U_2	D_0		D_0
U_3	D_0	D_0	

When $|D_0| = 3$, there are only four possible NAC structures fulfilling Definition 3.1:

$$\begin{aligned}
\big\{\{U_1, U_2, U_3\}, \{U_1\}, \{U_2\}, \{U_3\}\big\} &= \text{structure } \Lambda_{123};\\
\big\{\{U_1, U_2, U_3\}, \{U_2, U_3\}, \{U_1\}, \{U_2\}, \{U_3\}\big\} &= \text{structure } ; \lambda_{23}\\
\big\{\{U_1, U_2, U_3\}, \{U_1, U_2\}, \{U_1\}, \{U_2\}, \{U_3\}\big\} &= \text{structure } ; \lambda_{12}\\
\big\{\{U_1, U_2, U_3\}, \{U_1, U_3\}, \{U_1\}, \{U_2\}, \{U_3\}\big\} &= \text{structure } .\lambda_{13}
\end{aligned}$$

The sca matrix given on the left-hand side above corresponds to λ_{23}, while the sca matrix given on the right-hand side above corresponds to Λ_{123}, the trivial trivariate structure. As there are only four possible structures when $|D_0| = 3$, there are also only four 3-by-3 genuine sca matrices.

6. Sufficiency of structures on triples

Let λ be a NAC structure on a finite set $D_0 = \{U_1, ..., U_d\}$, $d \geq 4$. Suppose that for every triple $K_{ijk} = \{U_i, U_j, U_k\}$ with distinct $i, j, k \in \{1, ..., d\}$, the 3×3 sca matrix of $\lambda \sqcap K_{ijk}$, the tree spanned on $\{U_i, U_j, U_k\}$, is known. The set of 3×3 sca matrices built this way includes a total of $\binom{d}{3}$ sca matrices and will be referred to as ${}^3(\lambda)$.

In Proposition 6.1, it is shown that the NAC structure λ can be recovered from ${}^3(\lambda)$. Lemmas 1 to 3 contain some auxiliary results.

LEMMA 1. *Let λ be a NAC structure on $D_0 = \{U_1, ..., U_d\}$. For $\varnothing \neq T \subset C \subset D_0$, we have*

$$\operatorname{sca}(T, \lambda \sqcap C) = \operatorname{sca}(T, \lambda) \cap C.$$

Proof. By definition, we have

$$\operatorname{sca}(T, \lambda) \cap C = \Bigg(\bigcap_{B \in \lambda : T \subseteq B} B \Bigg) \cap C = \bigcap_{B \in \lambda : T \subseteq B} (B \cap C).$$

Since T is a subset of C and since T must be a subset of B, notice that requiring $T \subset B$ is equivalent in requiring $T \subset B \cap C$. Thus we can write

$$\mathrm{sca}(T, \lambda) \cap C = \bigcap_{B \in \lambda : T \subseteq B \cap C} (B \cap C).$$

On the other hand,

$$\mathrm{sca}(T, \lambda \sqcap C) = \bigcap_{B' \in \lambda \sqcap C : T \subseteq B'} B'.$$

Since $\lambda \sqcap C = \{B \cap C : B \in \lambda\} \setminus \{\varnothing\}$ by definition, we can rewrite the above expression as

$$\mathrm{sca}(T, \lambda \sqcap C) = \bigcap_{B \in \lambda : T \subseteq B \cap C, B \cap C \neq \varnothing} (B \cap C).$$

And because $T \subseteq B \cap C$ and $T \neq \varnothing$, the requirement $B \cap C \neq \varnothing$ can be dropped, thus

$$\mathrm{sca}(T, \lambda \sqcap C) = \bigcap_{B \in \lambda : T \subseteq B \cap C} (B \cap C) = \mathrm{sca}(T, \lambda) \cap C.$$

□

Lemma 2. *Let λ be a tree on D_0. For any nonempty subsets T_1, T_2, C of D_0 such that $T_1 \cup T_2 \subset C$, we have*

$$\begin{aligned} & \mathrm{sca}(T_1, \lambda) = \mathrm{sca}(T_2, \lambda) \\ \iff & \mathrm{sca}(T_1, \lambda \sqcap C) = \mathrm{sca}(T_2, \lambda \sqcap C). \end{aligned}$$

Proof. By Lemma 1, we have

$$\mathrm{sca}(T_j, \lambda \sqcap C) = \mathrm{sca}(T_j, \lambda) \cap C \text{ with } j = 1, 2.$$

Suppose first $\mathrm{sca}(T_1, \lambda) = \mathrm{sca}(T_2, \lambda)$. We therefore have

$$\begin{aligned} \mathrm{sca}(T_1, \lambda \sqcap C) &= \mathrm{sca}(T_1, \lambda) \cap C \\ &= \mathrm{sca}(T_2, \lambda) \cap C \\ &= \mathrm{sca}(T_2, \lambda \sqcap C). \end{aligned}$$

On the other hand, suppose that $\mathrm{sca}(T_1, \lambda \sqcap C) = \mathrm{sca}(T_2, \lambda \sqcap C)$. Obviously,

$$\mathrm{sca}(T_1, \lambda) \supset \mathrm{sca}(T_1, \lambda) \cap C$$

and since T_2 is both a subset of $\mathrm{sca}(T_2, \lambda)$ and of C, we also have

$$\mathrm{sca}(T_2, \lambda) \cap C \supset T_2.$$

Because $\mathrm{sca}(T_1, \lambda \sqcap C) = \mathrm{sca}(T_2, \lambda \sqcap C)$ implying by Lemma 1 that $\mathrm{sca}(T_1, \lambda) \cap C = \mathrm{sca}(T_2, \lambda) \cap C$, we have

$$\mathrm{sca}(T_1, \lambda) \supset T_2,$$

which means that $\mathrm{sca}(T_1, \lambda)$ is an ancestor of T_2, but not necessarily the smallest. Therefore $\mathrm{sca}(T_1, \lambda) \supset \mathrm{sca}(T_2, \lambda)$. The converse inclusion holds as well, by symmetry of the argument. We conclude that the two sets $\mathrm{sca}(T_1, \lambda)$ and $\mathrm{sca}(T_2, \lambda)$ are in fact equal. □

Lemma 3. *Let λ be a tree on D_0 and let $A \in \lambda$. Let B be a nonempty subset of D_0 with a least two elements. The smallest common ancestor of B is equal to A if and only if $B \subset A$ and there exist distinct children B_1 and B_2 of A such that $B \cap B_1 \neq \varnothing$ and $B \cap B_2 \neq \varnothing$.*

Proof. Suppose first that A is the smallest common ancestor of B. Clearly $B \subset A$. Let $B_1, \ldots, B_p$ be the children of A and recall these children form a partition of A. Hence $B = B \cap A = \bigcup_{j=1}^{p} (B \cap B_j)$, and thus at least one of these intersections is not empty. However, if only one of these intersections would be nonempty, say $B \cap B_1$, then we would get $B = B \cap B_1$ and thus $B \subset B_1$, meaning that B_1 is also common ancestor of all elements of B. Since B_1 is a proper subset of A, this would be in contradiction with the assumption that A is the smallest common ancestor of B. Therefore if A is the sca of B, B has a nonempty intersection with a least two children of A.

Conversely, suppose that $B \subset A$ and that there exist distinct children B_1 and B_2 of A having nonempty intersections with B. Let A' be a node in λ such that $B \subset A'$. Then also $B \cap B_1 \subset A'$, and thus, as $B \cap B_1$ is nonempty, $A' \cap B_1$ is not empty. Similarly, $A' \cap B_2$ is not empty. Since B_1 and B_2 are disjoint, requirement (iii) in Definition 3.1 then forces B_1 and B_2 to be descendants of A'. As a consequence $A \subset A'$. We have obtained that A is included in every node A' containing B as a subset. We conclude that A is the smallest common ancestor of the elements of B, as required. □

Proposition 6.1. *The NAC structure λ can be recovered from the set $^3(\lambda)$, that is, it is possible to retrieve the partition of the set of pairs $\{U_i, U_j\}$ of D_0 into equivalence classes from the set $^3(\lambda)$.*

Proof. If two distinct pairs have an element in common, their union is a triple. The equivalence of the two pairs can then be decided from that triple. Indeed, let $\{U_i, U_j\}$ and $\{U_i, U_k\}$ be two pairs with exactly one element, U_i, in common. To see whether they have the same smallest common ancestor in λ, it is sufficient to consider the tree induced by λ on the triple $\{U_i, U_j, U_k\}$: it is known from Lemma 2 that the pairs $\{U_i, U_j\}$ and $\{U_i, U_k\}$ have the same smallest common ancestor in λ if and only if they have the same smallest common ancestor in $\lambda \sqcap \{U_i, U_j, U_k\}$.

However, if two pairs are disjoint, there is no triple containing both pairs. Still, considering triples turns out to be sufficient to verify their equivalence:

the two pairs can only be equivalent if there is a third pair equivalent to both of them and having a non-empty intersection with each of them. Indeed suppose there exists a pair $\{U_i, U_j\}$ having the same smallest common ancestor as the pair $\{U_i, U_k\}$. Also suppose $\{U_i, U_k\}$ has the same smallest common ancestor as $\{U_k, U_l\}$. Then by transitivity $\{U_i, U_j\}$ has the same smallest common ancestor as $\{U_k, U_l\}$. Conversely, suppose that $\{U_k, U_l\}$ and $\{U_i, U_j\}$ have the same smallest common ancestor, A. Recall Lemma 3. Let B_i, B_j, B_k, B_l be the children of A to which U_i, U_j, U_k, U_l belong, respectively. We must have $B_i \cap B_j = \varnothing$ and $B_k \cap B_l = \varnothing$. Then B_k and B_l cannot both be equal to B_i.

- If B_k is different from B_i, then U_i and U_k belong to two different children of A, and the smallest common ancestor of $\{U_i, U_k\}$ is A too;
- If B_l is different from B_i, then, similarly, the smallest common ancestor of $\{U_i, U_l\}$ is A too.

In both cases, we have found a pair that is equivalent to $\{U_i, U_j\}$ and $\{U_k, U_l\}$ and that has a nonempty intersection with each of them. □

Hereafter is a practical example on how to retrieve λ from $^3(\lambda)$ for the case $d = 4$, $\binom{4}{3} = 4$.

Suppose the 3×3 sca matrices are:

$$\begin{array}{c|ccc} & U_1 & U_2 & U_3 \\ \hline U_1 & & H & I \\ U_2 & H & & I \\ U_3 & I & I & \end{array} \qquad \begin{array}{c|ccc} & U_1 & U_2 & U_4 \\ \hline U_1 & & K & L \\ U_2 & K & & L \\ U_4 & L & L & \end{array}$$

$$\begin{array}{c|ccc} & U_1 & U_3 & U_4 \\ \hline U_1 & & M & M \\ U_3 & M & & N \\ U_4 & M & N & \end{array} \qquad \begin{array}{c|ccc} & U_2 & U_3 & U_4 \\ \hline U_2 & & Q & Q \\ U_3 & Q & & R \\ U_4 & Q & R & \end{array}$$

From this, we get that

- The smallest common ancestors of the pair $\{U_1, U_2\}$ are $\{H, K\}$;
- The smallest common ancestors of the pair $\{U_1, U_3\}$ are $\{I, M\}$;
- The smallest common ancestors of the pair $\{U_1, U_4\}$ are $\{L, M\}$;
- The smallest common ancestors of the pair $\{U_2, U_3\}$ are $\{I, Q\}$;
- The smallest common ancestors of the pair $\{U_2, U_4\}$ are $\{L, Q\}$;
- The smallest common ancestors of the pair $\{U_3, U_4\}$ are $\{N, R\}$.

It appears therefore that $\{U_1, U_3\}, \{U_1, U_4\}, \{U_2, U_3\}$ and $\{U_2, U_4\}$ belong to the same equivalence class, while $\{U_1, U_2\}$ is all alone, as well as $\{U_3, U_4\}$. The branching nodes of λ in this case are therefore $\{U_1, U_2, U_3, U_4\}, \{U_1, U_2\}$ and $\{U_3, U_4\}$. The rooted tree structure λ is thus as shown in Figure 2.

The general procedure for any $d \geq 4$ is:

1. Establish a list of all possible pairs $\{U_i, U_j\}$, $i < j$;
2. For each pair, get from ${}^3(\lambda)$ the set of smallest common ancestors. Each pair should appear in $d-2$ trivariate sca matrices and thus $d-2$ smallest common ancestors should be available for each pair;
3. Intersect the set of smallest common ancestors of each pair with the sets of the other pairs. Any nonempty intersection means the two pairs are related, that is, belong to the same equivalence class;
4. Take the union of all pairs within each equivalence class to get the branching nodes of the structure;
5. Add the leaves to get λ.

7. Nonparametric estimation of a trivariate NAC structure

Let (X_1, X_2, X_3) be a vector of continuous random variables such that the joint distribution of $(U_1, U_2, U_3) = (F_{X_1}(X_1), F_{X_2}(X_2), F_{X_3}(X_3))$ is a nested Archimedean copula, and where F_{X_1}, F_{X_2} and F_{X_3} are the marginal CDFs of (X_1, X_2, X_3). We are interested in estimating the NAC structure based on n observations (x_{l1}, x_{l2}, x_{l3}) from (X_1, X_2, X_3), $l = 1, ..., n$.

Remember there are only four NAC structures possible for the trivariate case, as outlined at the end of Section 5. With the trivial structure (structure Λ_{123}), all marginal bivariate distributions of the NAC are the same while in structures λ_{23}, λ_{12} and λ_{13}, two marginal bivariate distributions are the same and one is different. Moreover if the marginal bivariate distributions are not all the same, being able to determine the one that is different from the two others is enough to determine whether the NAC structure is structure λ_{23}, λ_{12} or λ_{13}.

It is known from Genest and Rivest (1993) that the Kendall distribution of a pair of variables (X_j, X_k) fully determines the copula of that pair if it is an Archimedean copula. Thus, rather than working directly with bivariate distributions, let us work with the related Kendall distributions which are univariate and therefore easier to handle. The Kendall distribution of the pair $(X_j,\, X_k)$ is defined as the distribution of the variable

$$W_{jk} = C_{jk}(U_j, U_k) = H_{jk}(X_j, X_k)$$

where $C_{jk}(u_j, u_k) = P(U_j \leq u_j, U_k \leq u_k)$ is the joint CDF of (U_j, U_k) and where $H_{jk}(x_j, x_k) = P(X_j \leq x_j, X_k \leq x_k)$ is the joint CDF of (X_j, X_k). The map defined, for all $w \in [0, 1]$, by

$$K_{jk}(w) = P(W_{jk} \leq w)$$

is the Kendall distribution function (Barbe et al. 1996; Nelsen et al. 2003; Genest and Rivest 2001).

The Kendall distribution function of a pair of variables (X_j, X_k) can be estimated (Genest, Nešlehová and Ziegel, 2011) by first computing its pseudo-observations $w_{1,jk}, \ldots, w_{n,jk}$ and then by computing the empirical distribution function of these pseudo-observations:

$$w_{m,jk} = \frac{1}{n+1} \sum_{l=1}^{n} 1(x_{lj} < x_{mj}, x_{lk} < x_{mk});$$

$$F_{n,jk}(x) = \frac{1}{n} \sum_{m=1}^{n} 1(w_{m,jk} \leq x), \text{ with } 0 < x < 1.$$

Since there are three possible pairs in our case, namely $(X_1, X_2), (X_1, X_3)$ and (X_2, X_3), three empirical Kendall distribution functions can be estimated. A distance between the empirical Kendall distribution functions of (X_i, X_j) and (X_i, X_k) is defined as

$$\int_0^1 |F_{n,ij}(x) - F_{n,ik}(x)| \; dx = \frac{1}{n} \sum_{m=1}^{n} |w_{(m),ij} - w_{(m),ik}| = \delta_{ij,ik}$$

where $w_{(1),ij}, ..., w_{(n),ij}$ are the ordered pseudo-observations of the Kendall distribution related to the variables (X_i, X_j) and $w_{(1),ik}, ..., w_{(n),ik}$ are the ordered pseudo-observations of the Kendall distribution related to the variables (X_i, X_k).

Typically, a trivial structure will result in three distances that are all about the same, while structures such as λ_{12}, λ_{13} or λ_{23} will result in one small distance relative to two other distances that are bigger and about the same. Thus for any triple of variables (X_i, X_j, X_k), if, for instance, $\delta_{ij,ik}$ is the minimum among the three distances, it seems reasonable to assume that either the structure of the triple is the trivial structure or the structure λ_{jk} where (X_i, X_j) and (X_i, X_k) have the same Kendall distribution.

The problem of determining the structure of (X_1, X_2, X_3) can be rewritten as an hypothesis test:

H_0 : the true structure is the trivial structure.
H_1 : the true structure is structure λ_{12} or λ_{13} or λ_{23}, depending on what was the minimum observed distance.

As a test statistic, the absolute difference between the minimum distance and the average of the two remaining distances is used. The null hypothesis is rejected when the test statistic is observed in the upper tail of its H_0 distribution.

Unfortunately, the H_0 distribution of the test statistic is unknown. Under H_0 the original sample is assumed to come from an unknow trivariate Archimedean copula. Using the work of Genest et al. (2011), it is possible to estimate that Archimedean copula nonparametrically and to resample from that estimated AC, see Genest et al. (2011) for more details. For each new sample, the three empirical Kendall distributions, the three distances, and the related test statistic are to be computed. The p-value of the observed test statistic is then estimated by the proportion of test statistics obtained from the new samples that are greater than or equal to the value of the observed test statistic obtained from the original sample. Should this estimated p-value be lower than or equal to a threshold α, for instance 10%, the null hypothesis is to be rejected.

Note that the estimator for the Kendall distribution depends on the data only through the ranks and since our hypothesis test depends on this estimator, the resulting NAC structure estimator we developed here is rank-based too.

There are two key points in the test presented above:

- First, determine what should be the alternative hypothesis. Should it be structure λ_{12}, λ_{13} or λ_{23}?
- Second, choose between a trivial structure ($= H_0$) and H_1.

Possible errors are:

- If the true structure is the trivial structure, rejecting it and therefore committing a type I error;
- If the true structure is structure λ_{12}, λ_{13} or λ_{23}, failing to reject H_0 (type II error);
- If the true structure is for instance structure λ_{12}, getting a wrong H_1 and then picking it (we will call this a type III error).

The main difficulty with the test developed in this section is encountered when the true structure is the trivial structure, that is, the structure one gets when the nested Archimedean copula is actually a simple Archimedean copula. Indeed if the probability of committing a type I error is fixed to $\alpha = 0.10$, the trivial structure will be rejected 10% of the time regardless the input sample size n. Our estimator is therefore not a consistent estimator for the trivial trivariate structure, unless we let α tend to 0 as n increases, a key point if one hopes to achieve consistency for any trivariate NAC structure, including the trivial one.

8. Reconstruction of a NAC structure based on a set of estimated trivariate structures

Let λ be a NAC structure on a finite set $D = \{U_1, ..., U_d\}, d \geq 4$. It is known from Section 6 that if for every triple $K_{ijk} = \{U_i, U_j, U_k\}$ with different $i, j, k \in \{1, ..., d\}$ the 3×3 sca matrix of $\lambda \sqcap K_{ijk}$ is known, then λ can be recovered from that set of 3×3 sca matrices, this set of sca matrices being referred to as $^3(\lambda)$.

However if each of the 3×3 sca matrices are estimated, the problem is a bit different. Indeed it is not guaranteed that a proper NAC structure can be recovered from a given set of estimated 3×3 sca matrices. When the global estimated structure $\hat{\lambda}$ retrieved from $\widehat{^3(\lambda)}$ is not a genuine NAC structure, meaning it does not fulfill Definition 3.1, we call $\widehat{^3(\lambda)}$ a faulty set.

With a value of α equal to 0.00 for all tests, we fail to reject the null hypothesis everywhere and we therefore get a set of estimated sca matrices each describing a trivial trivariate structure. Such a set is never a faulty set, and $\hat{\lambda}$, the estimated global NAC structure retrieved from it, will always be a trivial structure of dimension d. Of course if the true structure is not a trivial structure of dimension d, a value of α equal to 0.00 means you are sure to commit type II errors.

With a value of α equal to 1.00 for all tests, all null hypotheses are rejected and we end up with a set where each 3×3 estimated sca matrix describes a non-trivial trivariate structure. Such a set can be a faulty set and usually is.

Assuming the copula of the vector $(X_1, ..., X_d)$ is a NAC, a faulty set of estimated trivariate structures means at least one error (type I, type II or type III) has been committed. Notice the converse is not true: even when at least one type I, type II or type III error has been committed, the set of estimated trivariate structures might lead to a global estimated NAC structure meeting Definition 3.1.

How to properly deal with a faulty set remains an open problem. As done in the simulation study, we simply suggest in such case to decrease the value of α for all tests till the resulting set of estimated trivariate structure is not a faulty set anymore. At worst, α is to be decreased down to 0.00, and we end up with a set of trivial trivariate structures. The global predicted structure is then the trivial structure of dimension d.

9. Simulation study

9.1. Testing the method with samples from a trivial trivariate structure

Let (U_1, U_2, U_3) be a vector of random variables having an Archimedean copula as joint distribution. We generate 500 samples of size n from this vector using the R package **nacopula**[2] (Hofert and Maechler, 2011). With $\alpha = 0.10$, how many times among the 500 samples are we able to retrieve the true structure? Figure 3 shows the percentage of correct predictions for various values of n, various generator families and two different values of the related parameter θ, expressed as Kendall's τ coefficient for convenience according to Table 1:

[2] The nacopula package has since been merged with the **copula** package.

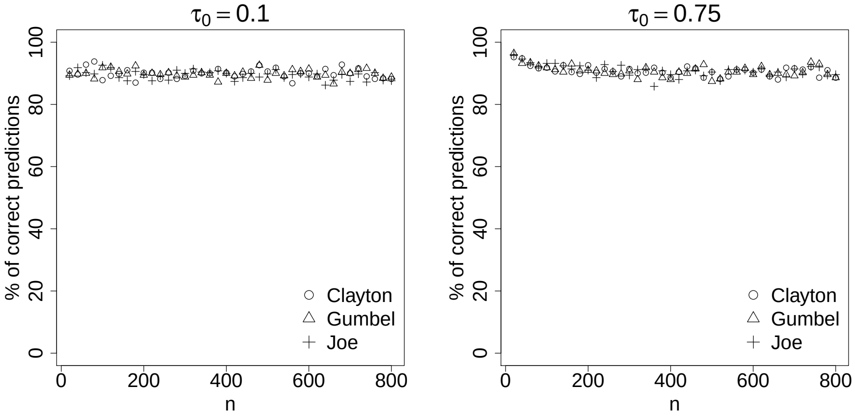

Figure 3: Percentage of correct predictions when the true structure is the trivial trivariate structure.

As expected, the percentage of correct predictions does not converge towards 100% but simply oscillates around 90%. In order for our estimator to be consistent for a trivial trivariate structure and therefore for any larger NAC structure that has at least one trivariate component equal to the trivial trivariate structure, we have to let α tend to 0 as n increases to ensure type I errors are asymptotically impossible.

To apply the method from Okhrin et al. (2013), we use the function **estimate.copula** of the R package HAC. As done in the simulation section of Okhrin and Ristig (2012b), we set **epsilon** to 0.15 for the aggregation step and use the default aggregation method. Since only the Clayton and Gumbel generator families are currently implemented in the HAC package and since the estimator from Okhrin et al. (2013) requires the knowledge of the generator family prior to the estimation of the structure, no comparison with the performances of our estimator for other families is possible at the time of writing.

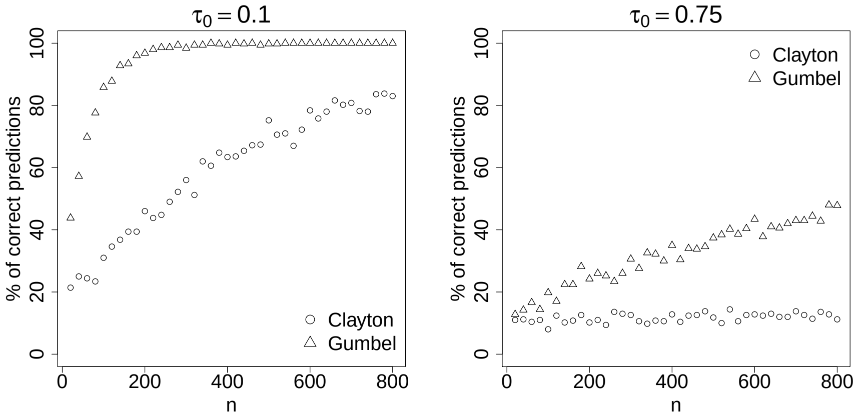

Figure 4: Performances of the estimator developed in Okhrin et al. (2013), with $\epsilon = 0.15$ and default aggregation method.

Increasing the value of ϵ improves the performances of their estimator in the case of a trivariate trivial structure, but decreases the performances of the same estimator when the target structure is a non-trivial trivariate structure. Later in this section, a seven-variate structure made up only of non-trivial trivariate structures is tested. For this last structure, a value of $\epsilon = 0.15$ is actually already too high and lead to poor performances of their estimator, thus preventing us from using a higher value of ϵ here (the same value of ϵ for their estimator is used throughout the simulation study, as well as the same value of α for our approach).

9.2. Testing the method with samples from a non-trivial trivariate structure

Given 500 samples of size n from a non-trivial trivariate structure, such as the one in Figure 1, and $\alpha = 0.10$, how many times among the 500 samples are we able to retrieve this non-trivial trivariate structure? Figure 5 shows the percentage of correct predictions for various values of n and various generator families. Note the same generator family is always used across all nodes of a given structure in the simulation section of this paper. The parameters θ_0 (root node, D_0) and θ_{23} (the other branching node, D_{23}) are expressed as Kendall's τ coefficients for convenience:

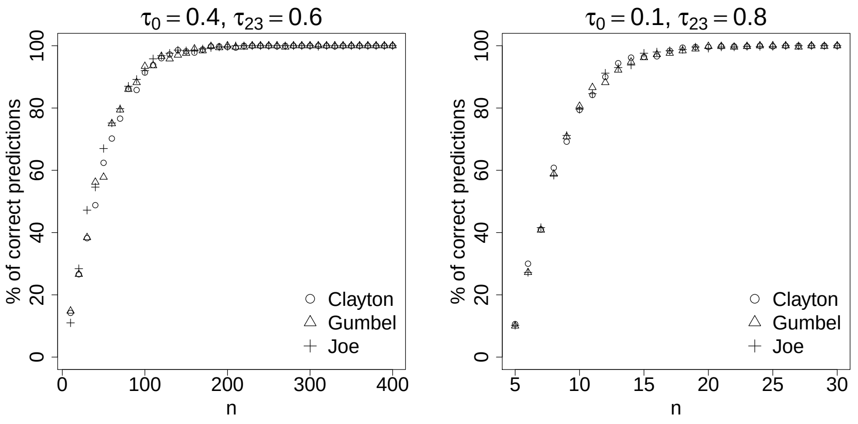

Figure 5: Percentage of correct predictions when the true structure is structure λ_{23}.

As the sample size increases, there is a clear convergence towards 100% of correct predictions. The more apart τ_0 and τ_{23}, the faster the convergence towards 100% of correct predictions (compare the two horizontal axes above). These results strongly suggest our estimator is a consistent estimator for any non-trivial trivariate NAC structure and thus for any larger NAC structure made up only of non-trivial trivariate structures.

Using the method from Okhrin et al. (2013) as we did in the previous subsection ($\epsilon = 0.15$, default aggregation method), we found out that the percentage of correct predictions also converges towards 100%, but at a much faster rate: their estimator clearly outperforms our estimator this time. The rate of convergence can be improved further by lowering the value of ϵ. However, recall that their estimator uses the knowledge of the generator family.

9.3. Testing the method with a four-variate structure

Suppose 500 samples of size n are generated from the structure below on the left-hand part of Figure 6, with $\tau_0 = 0.3$ and $\tau_{34} = 0.7$. With a fixed value of $\alpha = 0.10$, how many times are we able to retrieve this four-variate structure among the 500 samples? The right-hand part of Figure 6 shows the percentage of correct predictions for various values of n and various generator families:

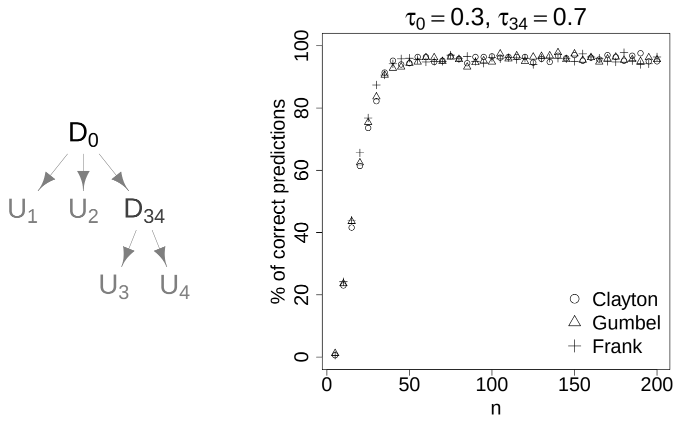

Figure 6: Percentage of correct predictions for a four-variate case.

The percentage of correct predictions eventually oscillates around 97%. There is no convergence towards 100%, which was expected for this structure since two of its trivariate components are trivial trivariate structures. Remark: in case the global predicted structure is not a genuine NAC structure, the value of α is decreased till the global predicted structure becomes a genuine NAC structure, possibly a trivial four-variate structure, refer to Section 8 for more details.

Using the method from Okhrin et al. (2013) as we did in the previous subsections ($\epsilon = 0.15$, default aggregation method), we found out that our estimator outperforms their estimator by a large amount. For instance, with a sample size $n = 75$, the number of correct predictions is 20% for the Clayton family and 50% for the Gumbel family versus 97% for all families tested with our estimator.

9.4. A seven-variate case

The performance of our method for a larger structure will be assessed by generating 500 samples of size n from the structure on the left-hand part of Figure 7, with $\tau_0 = 0.1$, $\tau_{123} = 0.3$, $\tau_{23} = 0.6$, $\tau_{4567} = 0.3$, $\tau_{567} = 0.5$ and $\tau_{67} = 0.8$. With a value of $\alpha = 0.10$, how many times are we able to retrieve this seven-variate structure among the 500 samples? The right-hand part of Figure 7 shows the percentage of correct predictions for various values of n and three generator families:

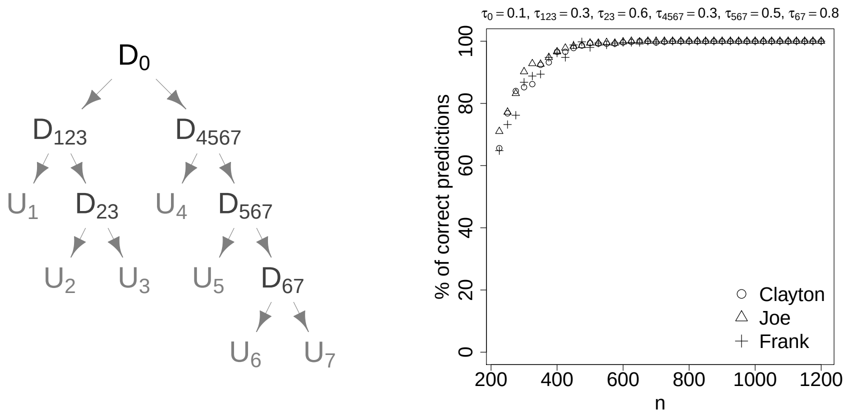

Figure 7: Percentage of correct predictions for a seven-variate case.

A convergence towards 100% of correct predictions can be observed. As there are no trivariate components equal to the trivial trivariate structure in the global structure, this was expected. As in the previous subsection, faulty structures are handled by decreasing α untill a valid structure emerges.

Using the method from Okhrin et al. (2013) as we did in the previous subsections ($\epsilon = 0.15$, default aggregation method), we found out that our estimator outperforms their estimator by a large amount. For instance, with a sample size $n = 1200$, the number of correct predictions of the Clayton family is barely 40% versus 100% for our estimator, and the convergence towards 100% of correct predictions is very slow. Lowering the value of ϵ however can help, for instance with a value of $\epsilon = 0$ (the smallest allowed), the number of correct predictions is 96% with a sample size as low as $n = 200$. However a value of ϵ equal to 0 for their estimator means that no aggregation is done anymore, making their estimator biased for many structures, for instance any trivial structure of dimension d.

10. Application

Daily log returns from January 2010 to December 2012 of

- Abercrombie & Fitch Co. (ANF), traded in New York,
- Amazon.com Inc. (AMZN), traded in New York,
- China Mobile Limited (ChM), traded in Hong Kong,
- PetroChina (PCh), traded in Hong Kong,
- Groupe Bruxelles Lambert (GBLB), traded in Brussels,
- and KBC Group (KBC), traded in Brussels,

were gathered with the help of Yahoo! Finance ($n = 700$ observations, $d = 6$). Figure 8 shows the estimated structure for ANF, AMZN, ChM and PCh, the estimated structure for ANF, AMZN, GBLB and KBC, and the estimated structure for ChM, PCh, GBLB and KBC.

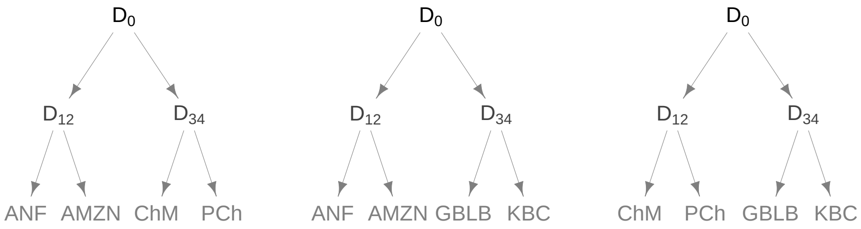

Figure 8: Given two log returns from one geographical area and two from another area, a natural clustering by area arises. The above structures are all strongly supported by the data, as the 12 related p-values are less than 10e-04.

In order to build a six-variate structure, we need to estimate the structure of eight extra triples. The left-hand panel of Figure 9 shows a reasonable guess for the six-variate structure in which the eight extra triples all have a trivial trivariate structure.

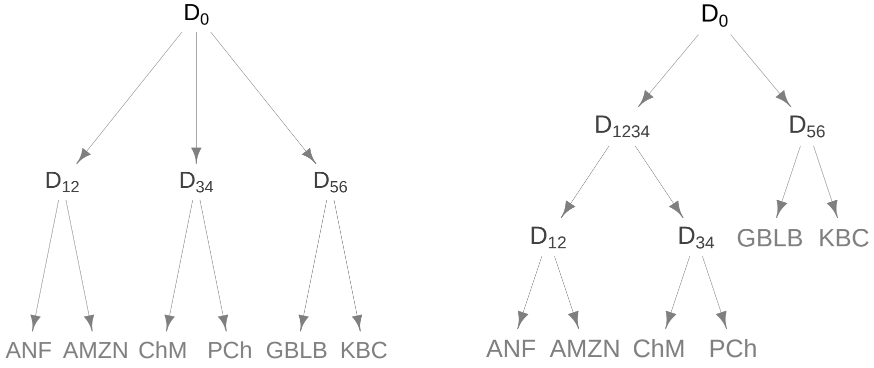

Figure 9: Possible six-variate structures for the data.

However, the trivial trivariate structure in four of the eight extra triples is strongly rejected by the data and suggest the structure in the right-hand of Figure 9. Unfortunately, this last structure implies we must reject the trivial trivariate structure for all eight extra triples and not only for half of them, making the prediction of a six-variate structure quite difficult. Since both PetroChina and China Mobile are traded not only in Hong Kong but also in New York, we could expect their log returns in Hong Kong to be more related to the log returns of some companies in New York (for instance ANF and AMZN) than to the log returns of two companies in Belgium. The structure on the right-hand of Figure 9 seems therefore more appropriate.

11. Discussion

In this paper, we have paved the way for a nonparametric rank-based approach to estimate a NAC structure, without any knowledge about the nested Archimedean copula prior to the estimation of its structure being necessary. A number of challenges however remain:

- Difficulties can appear when the method is applied to real data for which the true copula is not necessarily a NAC. For instance, one can end up with a subset of estimated non-trivial trivariate structures each strongly supported by the data (that is, very small p-values, meaning type I or type III errors are unlikely) and yet these non-trivial trivariate structures contradict each other in the sense that no global structure can be retrieved.

- Assuming the true copula of a sample is a NAC, being able to cope with a faulty set of estimated trivariate structures in a different way than the one suggested at the end of Section 8 and applied in the simulation section might result in better performances of the estimator, especially for small samples.

- The whole method is computationally intensive, unlike the method from Okhrin et al. (2013). This is best understood by calculating the number of triples for which a test is necessary with our approach: with $d = 10$, we indeed have to estimate 120 trivariate structures. With $d = 20$, this number increases to 1140. An optimized R code is available from the authors.

- Given an input sample of size $n \times d$, it is unclear how to determine what should be the optimal value for α. Asymptotically, one should have $\alpha_n \to 0$ if one hopes to have a consistent estimator for any NAC structure.

- Once a genuine NAC structure has been estimated, the problem of estimation of the generators remains. These generators cannot be estimated marginally, as doing so does not guarantee that the resulting function will be a proper copula.

References

References

Barbe, P., Genest, C., Ghoudi, K., Rémillard, B., 1996. On Kendall's process. Journal of Multivariate Analysis 58, 197–229.

Genest, C., Nešlehová, J., Ziegel, J., 2011. Inference in multivariate Archimedean copula models. Test 20, 223–256.

Genest, C., Rivest, L., 2001. On the multivariate probability integral transformation. Statistics & probability letters 53, 391–399.

Genest, C., Rivest, L.P., 1993. Statistical inference procedures for bivariate Archimedean copulas. Journal of the American Statistical Association 88, pp. 1034–1043.

Hering, C., Hofert, M., Mai, J., Scherer, M., 2010. Constructing hierarchical Archimedean copulas with Lévy subordinators. Journal of Multivariate Analysis 101, 1428–1433.

Hofert, J., 2010. Sampling Nested Archimedean Copulas: With Applications to CDO Pricing. Ph.D. thesis.

Hofert, M., Maechler, M., 2011. Nested Archimedean Copulas Meet R: The nacopula Package. Journal of Statistical Software 39, 1–20. Please note the package nacopula has been merged with the package copula.

Hofert, M., Pham, D., 2012. Densities of nested Archimedean copulas. arXiv preprint arXiv:1204.2410 .

Joe, H., 1997. Multivariate Models and Dependence Concepts. Chapman and Hall, London.

McNeil, A.J., 2008. Sampling nested Archimedean copulas. Journal of Statistical Computation and Simulation 78, 567–581. doi:10.1080/00949650701255834.

McNeil, A.J., Nešlehová, J., 2009. Multivariate Archimedean copulas, d-monotone functions and l_1-norm symmetric distributions. ArXiv e-prints arXiv:0908.3750.

Nelsen, R., Quesada-Molina, J., Rodríguez-Lallena, J., Úbeda-Flores, M., 2003. Kendall distribution functions. Statistics & probability letters 65, 263–268.

Okhrin, O., Okhrin, Y., Schmid, W., 2013. On the structure and estimation of hierarchical archimedean copulas. Journal of Econometrics 173, 189 – 204. doi:10.1016/j.jeconom.2012.12.001.

Okhrin, O., Ristig, A., 2012a. HAC: Estimation, simulation and visualization of Hierarchical Archimedean Copulae (HAC). R package version 0.2-6.

Okhrin, O., Ristig, A., 2012b. Hierarchical Archimedean Copulae: The HAC Package. SFB 649 Discussion Papers SFB649DP2012-036. Sonderforschungsbereich 649, Humboldt University, Berlin, Germany.

Puzanova, N., 2011. A hierarchical archimedean copula for portfolio credit risk modelling. Deutsche Bundesbank Discussion Paper, Series 2.