The University of Sydney Business School
The University of Sydney

BUSINESS ANALYTICS WORKING PAPER SERIES

Estimation of Hierarchical Archimedean Copulas as a Shortest Path Problem

Dmytro Matsypura, Emily Neo, Artem Prokhorov

Abstract

We formulate the problem of finding and estimating the optimal hierarchical Archimedean copula as an amended shortest path problem. The standard network flow problem is amended by certain constraints specific to copulas, which limit scalability of the problem. However, we show in dimensions as high as twenty that the new approach dominates the alternatives which usually require recursive estimation or full enumeration.

JEL Classification: C13

Keywords: network flow problem, copulas

April 2016

BA Working Paper No: BAWP-2016-05
http://sydney.edu.au/business/business_analytics/research/working_papers

Estimation of Hierarchical Archimedean Copulas as a Shortest Path Problem

Dmytro Matsypura[*] University of Sydney

Emily Neo[†] University of Sydney

Artem Prokhorov[‡] University of Sydney

April 12, 2016

Abstract

We formulate the problem of finding and estimating the optimal hierarchical Archimedean copula as an amended shortest path problem. The standard network flow problem is amended by certain constraints specific to copulas, which limit scalability of the problem. However, we show in dimensions as high as twenty that the new approach dominates the alternatives which usually require recursive estimation or full enumeration.

JEL Classification: C13

Keywords: network flow problem, copulas

[*]The University of Sydney Business School; E-mail: dmytro.matsypura@sydney.edu.au

[†]The University of Sydney Business School; E-mail: eneo3177@uni.sydney.edu.au

[‡]The University of Sydney Business School; E-mail: artem.prokhorov@sydney.edu.au

1 Introduction

An Archimedean n-copula is a copula of the following form

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

where $\psi : [0,1] \to [0,\infty]$ is a continuous generator function with first and second derivatives satisfying $\psi'(u) < 0$ and $\psi''(u) > 0$ for all $u \in (0,1)$. The generator functions are usually parameterized using a single parameter θ, e.g., if $\psi(u) = (-\ln u)^\theta$ then we obtain the Gumbel copula, if $\psi(u) = -\ln \frac{e^{-\theta u}-1}{e^{-\theta}-1}$ we have the Frank copula. A hierarchical, or nested, Archimedean n-copula (n-HAC) is obtained by using lower dimensional Archimedean copulas as arguments of lower dimensional Archimedean copulas so that the resulting object is n-dimensional (see, e.g., Joe, 1994). For example, a 3-HAC can be obtained using an Archimedean 2-copula as an argument of another Archimedean 2-copula as follows

$$C(u_1, u_2, u_3) = C\left(u_1, C(u_2, u_3; \psi_2); \psi_1\right). \tag{1}$$

Note that if the same generator function is used in all levels of the hierarchy the n-HAC trivially reduces to an Archimedean n-copula so HACs are a more general class.

HACs are not exchangeable and provide a much higher flexibility in modelling complex high-dimensional dependence. However, there is a large number of alternative hierarchies. In (1), we have used variable u_1 in the first level of the hierarchy, and variables (u_2, u_3) in the second, deeper, level but we could have picked any other order. Estimation of a HAC requires deciding on the optimal hierarchy, that is on the optimal level for each variable, as well as evaluating the generator function parameters. Because of the number of hierarchies to consider, this is not a trivial estimation task even in modest dimensions.

For an n-HAC to be a proper n-copula, its generator function has to satisfy a monotonicity property which amounts to having derivatives of all orders with alternating signs (see, e.g., Embrechts et al., 2003, p. 374). For commonly used generator functions including those we use in the paper, this requirement translates into a restriction on θ's. Specifically, if we denote by $\theta_j, j = 1, \ldots, n-1$, the parameter used in level j, where $j = 1$ corresponds to the outer-most level and $j = d-1$ corresponds to the inner most level, then they must satisfy the property that $\theta_1 < \theta_2 < \ldots < \theta_{n-1}$ (see, e.g., Joe, 1997, p. 88). This property – often called *the nesting condition* – further complicates estimation.

We propose taking a network approach, namely we determine the optimal structure as a so-

lution to an *amended* shortest path problem. The approach is appealing because for the standard formulations of an SPP there exist a large number of very efficient computational algorithms.

2 The Network Approach to HAC Estimation

Generally speaking, the problem of determining the correct HAC structure is of combinatorial nature. It can be thought of as the problem of selecting a correct structure from the set of possible structures. Obviously, as the number of variables grows, complete enumeration quickly becomes intractable due to a very large number of alternative hierarchies. For example, for $d = 10$, the number of alternative hierarchies is on the order of $10! \approx 3.6 \cdot 10^6$.

Hence, we are interested in a method that does not require complete enumeration. The proposed approach is based on the classical shortest path problem formulated as follows. Suppose that we are given a network G having m nodes, n arcs, and a distance d_{ij} associated with each arc (i, j) in G. Our goal is to find the shortest path from node s (source) to node t (sink) in G. The length of the path is the sum of the distances on the arcs in the path. The corresponding mathematical formulation is:

$$\text{Minimize} \quad \sum_{i=1}^{m}\sum_{j=1}^{m} d_{ij}x_{ij} \tag{2}$$

$$\text{subject to} \quad \sum_{j=1}^{m} x_{ij} - \sum_{k=1}^{m} x_{ki} = \begin{cases} 1 & \text{if } i = s \\ 0 & \text{if } i \neq s \text{ or } t \\ -1 & \text{if } i = t \end{cases} \tag{3}$$

$$x_{ij} = 0 \text{ or } 1 \qquad i, j = 1, \ldots, m, \tag{4}$$

where the sums and 0-1 requirements are taken over existing arcs in G and constraint (3) is the *conservation of flow* constraint, which ensures that the flow may neither be created nor destroyed in the network (Bazaraa et al., 2010). In the context of HACs, a measure inversely related to the strength of dependence represents the 'distance' between variables.

For example, consider the network in Fig. (1). Node v_{ij}, $i, j = 1, ..., n$, represents variable i used in level j of the HAC hierarchy. That is, v_{21}, for example, means that the second (out of n) variable is used in the first level of the HAC hierarchy and v_{12} means that the first variable is used in the second level. Aside from the starting and terminal nodes, there are n columns of nodes, each representing a level. In the inner most level we have two variables corresponding to the last two columns.

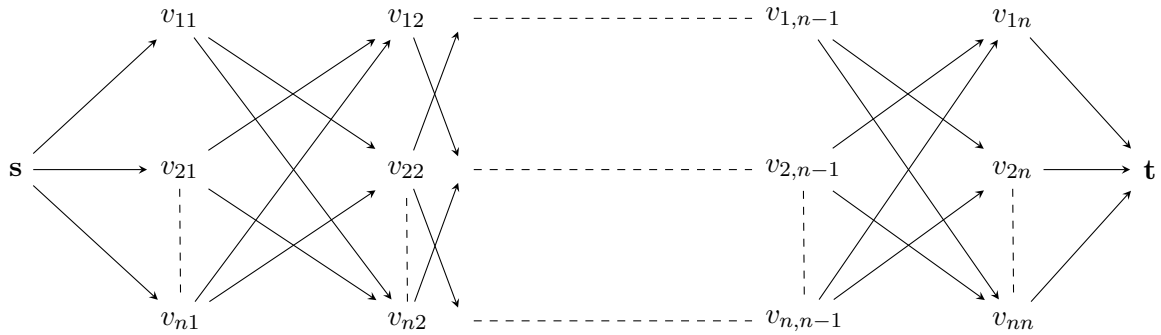

Figure 1: Capturing dependency as a directed shortest path problem.

There is a single variable per level except for the last two columns, which, by convention, jointly represent a single, inner-most, level. These structures are often called *fully nested*. The arcs between nodes represent dependence and the shortest path from s to t maximizes the aggregate dependency across arcs. Clearly, once a variable is used in the hierarchy it cannot be used again. This will impose another constraint in addition to the nesting constraints referred to in the Introduction.

Specifically, let $d_{ij,kl}$ denote the distance measure for variables indexed i and k, which are used in nesting levels j and l, respectively. Let $f_{ij,kl}$ denote a binary indicator for whether to go through that arc or not. The SPP we consider is similar to the classical formulation presented earlier but has the additional constraints, specific to HACs. Hence we call it an *amended* SPP. It can be stated as follows.

$$\text{Minimize} \quad \sum_{\forall(ij,kl)} d_{ij,kl} f_{ij,kl} + \sum_{j=1}^{n-1} s_j \tag{5}$$

$$\text{subject to} \quad \sum_{kl \in \mathbb{O}(ij)} f_{ij,kl} - \sum_{kl \in \mathbb{I}(ij)} f_{kl,ij} = \begin{cases} 1 & \text{if } ij = s \\ 0 & \text{if } ij \neq s \text{ or } t \\ -1 & \text{if } ij = t \end{cases} \tag{6}$$

$$\sum_{j} \sum_{kl} f_{ij,kl} = 1 \qquad \forall i \tag{7}$$

$$f_{ij,kl} = 0 \text{ or } 1 \qquad \forall ij, kl \tag{8}$$

$$\sum_{kl \in \mathbb{I}(ij)} d_{kl,ij} f_{kl,ij} - \sum_{kl \in \mathbb{O}(ij)} d_{ij,kl} f_{ij,kl} + s_j \leq 0 \qquad \forall j \tag{9}$$

For every node ij, the sets $\mathbb{I}(ij)$ and $\mathbb{O}(ij)$ represent the set of arcs entering ij and the set of arcs leaving ij, respectively. Constraint (7) ensures that each variable in the HAC is used exactly once. This is the additional constraint, specific to our problem.

The nesting constraints that the child nodes, or inner nested levels, must have larger parameter

values than their parent nodes, or outer levels, can be stated as follows

$$\sum_{kl \in \mathbb{I}(ij)} d_{kl,ij} f_{kl,ij} \leq \sum_{kl \in \mathbb{O}(ij)} d_{ij,kl} f_{ij,kl} \tag{10}$$

Unfortunately, adding this set of constraints tends to make the problem infeasible (due to $d_{ij,kl}$ being estimated). In order to overcome this issue we relax these constraints by adding slack variables $s_j \geq 0$ in constraint (9) and objective function (5). As this is a minimization problem, the algorithm always attempts to find a solution with the smallest sum of s_j's. Thus, if the amended SSP is feasible, the optimal solution obtained using the objective function in (5) is equivalent to the optimal solution obtained using the objective function without the term $\sum_{j=1}^{n-1} s_j$.

There are several options for the distance measure $d_{ij,kl}$. The most obvious is to estimate the θ that represents dependence between variable i in level j and the copula from level $j+1$. However this means that for every variable in level j we have to recursively estimate $(n-j-1)$-dimensional copulas for all arrangements of $n-j$ variables. There are $(n-j-1)!$ such copulas, assuming the inner most copula is exchangeable, so this approach would result in complete enumeration. Instead we propose to construct $d_{ij,kl}$ using $\hat{\theta}_{ik}$, where $\hat{\theta}_{ik}$ represents dependence between only two variables, i and k. Specifically, let $\hat{\theta}_{ik}$ be an MLE of the parameter in the generator function used in a bivariate copula connecting variables i and k. Then, $d_{ij,kl}$ can be any measure inversely related to dependence strength. One possibility is $d_{ij,kl} = \left(M - \hat{\theta}_{ik}\right)$, where M is some large number. This is the distance measure we use.

Although the inclusion of constraints (7) and the binary nature of $f_{ij,kl}$ make this a mixed integer programming (MIP) problem and break the structure of a standard SPP, these constraints are necessary in the context of HAC estimation. As an MIP problem, our formulation has no polynomial time solution algorithm known to us. This limits scalability of the problem. Yet, our approach exhibits a very promising behavior in large problems as compared with available alternatives. Moreover, the SPP representation considers all possible combinations of variable-level pairs, so it evaluates the entire HAC structure. Available competitors, on the other hand, tend to evaluate HACs level-by-level and to estimate HAC recursively.

3 Other Approaches

The two most popular approaches to estimating HACs available in the literature are the recursive maximum likelihood estimator (RMLE) of Okhrin et al. (2013) and the diagonal maximum likelihood estimator of Górecki et al. (2014). We use these estimators in our simulations.

3.1 RMLE

The recursive nature of RMLE stems from a consecutive estimation of bivariate copula parameters. This estimator proceeds as follows. First, we consider all pairs of variables and find the pair that has the largest copula parameter estimate obtained using MLE. This pair serves as the inner most level of the HAC hierarchy. Then, the copula estimate for that pair is used as one of the two marginals and we consider all combinations of that marginal with other variables. Again, we obtain a set of MLE estimates of the copula parameters and look for the largest value. This is the next level of the HAC hierarchy. We proceed in this way until we construct the entire HAC.

Okhrin et al. (2013) provide some asymptotic results for the RMLE, however Górecki et al. (2014) show that the RMLE is not consistent in general and provide a correction that ensures consistency. The RMLE procedure quickly becomes computationally demanding as the copula dimension grows but has been shown to have acceptable performance in simulations for low-dimensional problems.

3.2 DMLE

The DMLE corrects for the fact that the copula of a Kendall transformation of two variables (the copula-based marginal used in the RMLE) and a third variable is not in general equal to the copula of the three variables. For example, if we wish to model $C_0(u_1, C_1(u_2, u_3))$ and let K denote the Kendall transformation of vector (U_2, U_3) then vector (U_1, K) will not in general have the copula $C_0(\cdot, \cdot)$ as the distribution function. This means that the MLE of the copula parameter in C_0 will in general be inconsistent. A solution proposed by Górecki et al. (2014) is to use a corrected version of the RMLE where instead of the Kendall transformation one uses the diagonal of the relevant copula. More specifically, instead of K we would use a transformation defined as follows

$$\delta = \psi(2\psi^{[-1]}(\max\{U_2, U_3\})),$$

where ψ is the generator function of C_1 (see Górecki et al., 2014, for more details).

4 Simulation

We report simulation experiments comparing the network approach with RMLE and DMLE for dimensions equal to 5, 10 and 20. Table 1 reports simulation results for selected copulas from the Gumbel, Clayton and Frank Archimedean families.

The performance criteria we use are the integrated mean square error (IMSE), designed to

	RMLE		DMLE		Network		
	Structure	**IMSE**	**Structure**	**IMSE**	**Structure**	**IMSE**	**IMSE-δ**
			5 Dimensions				
Gumbel	100%	0.302	100%	0.401	100%	0.411	0.401
Clayton	100%	0.044	100%	0.060	100%	0.065	0.060
Frank	100%	0.495	100%	0.756	99%	0.845	0.766
			10 Dimensions				
Gumbel	-	-	100%	2.773	100%	2.250	2.773
Clayton	-	-	93%	0.015	97%	0.012	0.015
Frank	-	-	83%	0.265	78%	0.256	0.267
			20 Dimensions				
Gumbel	-	-	64%	2.314	62%	2.372	2.319
Clayton	-	-	48%	0.020	47%	0.017	0.019
Frank	-	-	18%	0.009	24%	0.009	0.009

Table 1: IMSE ($\times 10^{-5}$) and proportion of correctly estimated structures for the network methods and for RMLE and DMLE.

capture the average difference between the estimated copula function and the true function used for sampling, and the fraction of correctly uncovered structures. In simulating from Archimedean copulas we use the sampling procedure of Hofert (2011) and the number of points over which we evaluate estimators' performance is 10,000. We report the proportion of correctly estimated structures produced by each method over simulations. The number of simulations is 100, and the sample size generated is 1,000.

We start by using our approach for structure determination. Then we evaluate the resulting HAC using two estimators. One uses $\hat{\theta}_{ik}$'s to build the entire HAC as follows

$$\hat{C}\left(u_{i_{(1)}}, u_{i_{(2)}}, \ldots, u_{i_{(n)}}\right) = \hat{C}_{i_{(1)}\, i_{(2)}}\left(u_{i_{(1)}}, \hat{C}_{i_{(2)}\, i_{(3)}}\left(u_{i_{(2)}}, \hat{C}_{i_{(3)}\, i_{(4)}}\left(\ldots, \hat{C}_{i_{(n-1)}\, i_{(n)}}\left(u_{i_{(n-1)}}, u_{i_{(n)}}\right)\right)\ldots\right)\right),$$

where $\hat{C}_{i_{(p)}\, i_{(p+1)}}, p = 1, \ldots, n-1$, is the estimated bivariate copula for variables that happen to be used in level p. For example, if variables j and k were chosen to form level 1 then $\hat{C}_{i_{(1)}\, i_{(2)}}(\cdot,\cdot) = \hat{C}_{jk}(\cdot,\cdot) = C(\cdot,\cdot;\psi_{jk})$, where generator ψ_{jk} uses the estimate $\hat{\theta}_{jk}$.

The other estimator uses the above mentioned observation made by Górecki et al. (2014) and constructs the entire HAC as follows

$$\tilde{C}\left(u_{i_{(1)}}, u_{i_{(2)}}, \ldots, u_{i_{(n)}}\right) = \hat{C}_{i_{(1)}\, i_{(2)}}\left(u_{i_{(1)}}, \hat{\delta}_{i_{(2)}\, i_{(3)}}\left(u_{i_{(2)}}, \hat{\delta}_{i_{(3)}\, i_{(4)}}\left(\ldots, \hat{\delta}_{i_{(n-1)}\, i_{(n)}}\left(u_{i_{(n-1)}}, u_{i_{(n)}}\right)\right)\ldots\right)\right),$$

where $\hat{\delta}_{i_{(p)}\, i_{(p+1)}}(\cdot,\cdot) = \psi_{i_{(p)}\, i_{(p+1)}}\left(2\psi^{[-1]}_{i_{(p)}\, i_{(p+1)}}(\max\{\cdot,\cdot\})\right), p = 2, \ldots, n-1$. For example, if variables k and l were chosen for level 2 then $\hat{\delta}_{i_{(2)}\, i_{(3)}}(\cdot,\cdot) = \hat{\delta}_{kl}(\cdot,\cdot) = \psi_{kl}\left(2\psi^{[-1]}_{kl}(\max\{\cdot,\cdot\})\right)$ and ψ_{kl} uses

$\hat{\theta}_{kl}$.

In Table 1, under *Structure* we report the fraction of correctly identified structures and the two estimation methods are referred to as *'Network IMSE'* and *'Network IMSE-δ'*. The R-code implementing the estimators is available from the authors' webpages; the R-code implementing the alternative estimators (RMLE and DMLE) uses the *copula* package (Kojadinovic and Yan, 2010).

Table 1 suggests that the network approach using either estimation method is particularly competitive for larger problems, when the RMLE becomes computationally infeasible and DMLE produces a larger error. It can be seen that although the error of the network approach increases with the dimension, the growth rate of the error is no larger than for DMLE. Interestingly, for the Frank copula both DMLE and the network estimators produce larger error in structure determination but smaller IMSE when dimension is increased. The RMLE was not operational beyond six dimensions, while DMLE and the network method produced consistent results, with the network methods sometimes outperforming DMLE in larger dimensions.

5 Concluding remarks

We proposed a new approach to estimating HACs, which is based on viewing HACs as networks and representing dependence as a network flow. Available estimators are based on a recursive MLE argument in which deeper levels of HACs is assumed to have stronger dependence and the estimation proceeds recursively using MLE of the copula as a marginal distribution for higher levels of a HAC. Contrary to this, we propose estimating the entire structure by looking for the path through the network which maximizes the sum of dependence measures between the network nodes where a node represents a variable with a given position in the HAC.

Shortest path problems are well studied in the operations research literature. A complication arising from the HAC context is the "no return" and "nesting" conditions. We compare the amended SPP estimator with alternatives and show that it remains operational in fairly high dimensions and, perhaps surprisingly, behaves at least as well (in terms of integrated error and number of correctly identified structures) as the recursive alternative that works.

References

Bazaraa, M., J. Jarvis, and H. Sherali (2010): *Linear Programming and Network Flows*, Wiley.

Embrechts, P., F. Lindskog, and A. McNeil (2003): "Modelling Dependence with Copulas

and Applications to Risk Management," in *Handbook of Heavy Tailed Distributions in Finance: Handbooks in Finance*, ed. by S. T. Rachev, Elsevier, chap. 8, 329–384.

Górecki, J., M. Hofert, and M. Holeňa (2014): "On the consistency of an estimator for hierarchical Archimedean copulas," in *32nd International Conference on Mathematical Methods in Economics*, ed. by J. Talašová, J. Stoklasa, and T. Talášek, Palacký University, Olomouc, 239–244.

Hofert, M. (2011): "Efficiently sampling nested Archimedean copulas," *Computational Statistics and Data Analysis*, 55, 57–70.

Joe, H. (1994): "Multivariate extreme-value distributions with applications in environmental data," *The Canadian Journal of Statistics*, 22, 47–64.

——— (1997): *Multivariate Models and Multivariate Dependence Concepts*, Chapman & Hall/CRC Monographs on Statistics & Applied Probability, Taylor & Francis.

Kojadinovic, I. and J. Yan (2010): "Modeling Multivariate Distributions with Continuous Margins Using the copula R Package," *Journal of Statistical Software*, 34, 1–20.

Okhrin, O., Y. Okhrin, and W. Schmid (2013): "On the structure and estimation of hierarchical Archimedean copulas," *Journal of Econometrics*, 173, 189 – 204.