

On the use of copulas to simulate multicriteria data

Jairo Cugliari¹ and Antoine Rolland² and Thi-Min-Tuy Tran³

Abstract. Several methods have been proposed in the past decades to deal with Multicriteria Decision Aiding (MCDA) problems. However, a comparison between these methods is always arduous as the number of datasets proposed in the literature is very low. One of the limitations of the existing datasets is that generally MCDA methods are dealing with very small sets of data; typically, a MCDA problem deals with a number of alternatives that does not exceed 20 or 30 and is often less. Therefore, it should be interesting to propose a way to simulate new data based on some existing dataset, i.e. taking into account the potential links that should exist between the criteria. We introduce in this paper the use of the statistical functions named copula to simulate such data. A practical way to use copula is proposed, and the quality of the obtained data is discussed.

1 Introduction

Multicriteria Decision Aiding (MCDA) studies aim at helping a Decision Maker (DM) to take (good) decisions. Many different models have been proposed since more than 50 years (see [3] or [7] for a survey), among others:

- utility-based approaches, using linear (MAUT [15], AHP [24]) or non-linear (Choquet integral [14]) aggregation functions
- outranking approaches, like ELECTRE [13] or PROMETHEE [8] methods
- mixed methods, like rule-based methods [20, 21] and others.

There is still a great increase of the number of very specific methods, or variants of existing methods, to be proposed. All these methods are always presented as very interesting and perfectly adapted to the situation. The fact is that it is very difficult to test and compare different methods described in the literature, as they often are dedicated to one specific situation. Even if the axiomatic foundations have been generally well studied (see [7] for a first approach), it is often difficult to realize which are the difference *in practice* between the results obtained by two different methods. Therefore, there is a lack of testing sets of data on which one can try the different methods. Several solutions have already been proposed to increase the possibility of benchmark between MCDA methods. We can cite the Decision Deck project which proposes a unified data standard for MCDA data [5], and a unified web services platform through DIVIZ [18]. We can cite also a companion paper [22] which aims at proposing a repository of real or fictitious datasets for MCDA situations.

But sometimes only very few data are available; for example, from an preference learning point of view, the dataset should be so limited

that it is too small to be divided into a test subset and a validation subset. Researches should also desire to have more data to test the proposed methods. There is then a need to be able to increase the size of the datasets through simulated data. Good practices in MCDA point out the fact, among others, that criteria should be as independent as possible [23]. But in real life the values taken by an alternative on different criteria are generally not totally independent. For example, if the DM is facing a problem like flat rental, she would like to select several flats to visit. Obviously, data like surface, price, or rooms number seem to be good criteria to decide which flat to visit. But these criteria are often linked: increasing the surface is greater increase also the chance to have more rooms; or the price is an increasing function of the surface, with respect to other criteria. Therefore, MCDA data cannot be independently well simulated. The problem is then to model the interaction between criteria in a plausible way. We propose to use a statistical approach to overcome this difficulty. Copula is a statistical tool which aims at modelling those interactions. Basically, a copula is a function that describe a multivariate distribution as a function of the marginal univariate distributions. We propose in this paper to use copulas to first model the interactions between criteria, and then to simulate new alternatives. We automatically learn the copula parameters from the actual dataset (used as training set) so as to generate new simulated data sets.

As far as we know, there is no work about the simulation of multicriteria data except a tentative using Bayesian network presented in [2].

In this paper we present a practical way to use copulas to simulate MCDA data inspiring from the work in [12]. In section 2 we introduce the copulas functions and quickly present the most well-known copulas families. In section 3 we first stand the hypothesis under which we worked. We then present a process to elicitate the parameters of the copulas following [1]. Finally, we show some numerical experiments we performed on available MCDA dataset in the literature.

2 Copulas

In this section we recall some basic notions about modeling dependency with copulas (see [19] for a more formal presentation of the subject). The basic construction bricks will be pair copula constructions (PCC) which are assembled together in a vine copula.

2.1 A brief introduction to copulas

In a nutshell a copula is a multivariate cumulative distribution function which has all its margins uniformly distributed on the unit interval. If $U_1, \dots, U_n; n \geq 2$ are random variables with uniform distribution in $[0, 1]$, then a copula $C : [0, 1]^n \mapsto [0, 1]$ satisfies

$$C(u_1, \dots, u_n) = P(U_1 \leq u_1, \dots, U_n \leq u_n) \quad (1)$$

¹ Lab. ERIC, Université Lyon 2, email: Jairo.Cugliari@univ-lyon2.fr

² Lab. ERIC, Université Lyon 2, email: Antoine.Rolland@univ-lyon2.fr

³ Lab. ERIC, Université Lyon 2, email: Thi-minh-thuy.Tran@etu.univ-lyon1.fr

A central result on copulas is Sklar's theorem [25] which allows one to represent any n -variate cumulative distribution function $F(x_1, \dots, x_n)$ of the random vector $\mathbf{X} = (X_1, \dots, X_n)$ as

$$F(x_1, \dots, x_n) = C(F(x_1), \dots, F(x_n)), \quad (2)$$

where $F(x_1), \dots, F(x_n)$ are the univariate marginal distribution functions of the vector \mathbf{X} . Moreover, this representation is unique if the marginals are absolutely continuous. A converse result is Nelsen's corollary [19] which identifies the copula from the joint and marginal distribution

$$C(u_1, \dots, u_n) = F(F^{-1}(x_1), \dots, F^{-1}(x_n)). \quad (3)$$

Intuitively, the probabilistic structure of the vector \mathbf{X} is the result of coupling the marginal behavior of the components of \mathbf{X} by means of the copula C which has intermediate practical implications. For example, from the observation of n independent and identical realizations $\mathbf{X}_1, \dots, \mathbf{X}_n$ of \mathbf{X} , one can estimate the joint multivariate distribution function F by estimating the marginals and identifying one copula function among the elements of known copula families (e.g. the elliptical or Archimedean classes among others [19]). If F is absolutely continuous, then we use the chain rule to write the density equivalent to equation (2)

$$f(x_1, \dots, x_n) = c(F_1(x_1), \dots, F_n(x_n))f_1(x_1) \dots f_n(x_n) \quad (4)$$

where the copula density function c is given by

$$c(u_1, \dots, u_n) = \frac{\partial^n C(u_1, \dots, u_n)}{\partial u_1 \dots \partial u_n} \quad (5)$$

The difficulty of this problem depends on the data dimension n . In the bivariate case, e.g. $n = 2$, only one pair-copula must be estimated and many solutions have been already proposed to do so (see for example [16, Chapter 5]). However, several of these approaches are not feasible in higher dimension spaces.

2.2 Pair-Copula Construction (PCC)

To avoid some problems that arise on high dimension datasets, [4] propose a pair-copula construction in order to decompose the multivariate joint density of X into a cascade of building blocks called pair-copula.

As before f is the joint density of X which is factorized (uniquely up to a relabeling of the elements of X) as

$$f(x_1, \dots, x_n) = f(x_n)f(x_{n-1}|x_n) \dots f(x_1|x_2, \dots, x_n). \quad (6)$$

Then, one can write each of the conditional densities on (6) using (4) recursively which yields on this general expression for a generic element X_i of X given a generic conditioning vector v

$$f(x_i|v) = c_{x_i, v_j|v_{-j}}(F(x_i|v_{-j}), F(v_j|v_{-j})) \times f(x_i|v_{-j}). \quad (7)$$

In last expression we use the notation v_j for the j -th element of v and v_{-j} for all the elements of v but v_j .

For example, let take three random variables X_1, X_2 and X_3 . We have the following decomposition:

$$f(x_1|x_2|x_3) = c_{12|3}(F(x_1|x_3), F(x_2|x_3)) \times f(x_1|x_3). \quad (8)$$

2.3 Vines copulas

Vines copulas have been proposed to classify alternatives factorization of (6) into a structured graphical model [4]. This construction allows highly flexible decompositions of the (possibly high) dimensional distribution of X because each pair-copula can be chosen independently from the others. The iterative decomposition provided by the PCC is then arranged into a set of linked trees (acyclic connected graph). Two special schemes are usually used: C-vines (canonical vines) and D-vines. In the former one, a dependent variable is identified and chosen to be the root of the tree. In the following tree, the dependence will be computed conditional on this first variable and so on. In the latter scheme, a variable ordering is chosen. Then on the first tree one models the dependence of each of the consecutive pairs of variables. The following tree will model the dependence of the remaining pairs, conditional on the those that were already modeled. See [1] for a more detailed exposition of this construction.

2.4 Simulation

Simulation of copula data (i.e. n -variate data with uniformly distributed marginals) can be done using the probability integral transform. It is convenient to define the h -function

$$h(x|v, \theta) = \frac{\partial^d C_{x, v_j|v_{-j}}(F(x|v_j), F(x|v_{-j}), |\theta)}{\partial F(v_j|v_{-j})}, \quad (9)$$

where θ is a parameter vector associated to the decomposition level. The h -function is the conditional distribution of x given v and we let $h^{-1}(u|v, \theta)$ be its inverse with respect to u , i.e. the inverse of the cumulative conditional distribution. The simulation for the vine is as follows. First sample n uniformly distributed random variables w_1, w_2, \dots, w_n . Then use the probability integral transform of the corresponding conditional distribution:

$$\begin{aligned} x_1 &= w_1, \\ x_2 &= F^{-1}(w_2|x_1), \\ x_3 &= F^{-1}(w_3|x_1, x_2), \\ &\dots \\ x_n &= F^{-1}(w_n|x_1, \dots, x_{n-1}). \end{aligned}$$

At each step, the computation of the inverse conditional distribution is made through the (inverse) h -function.

3 Numerical experiments

The aim of the data simulation is to obtain new fictitious data in accordance with a set of real data. The model (copula) parameters are automatically learned from the real dataset, and then the model is used to simulate new data. Ideally, the new fictitious data should be indiscernible from the real ones. We detail in the following sections the hypothesis on the real data that we make, then the simulation process and the way we can prove that we reach our objective of indiscernibility.

3.1 Hypothesis

The input data are a set of p alternatives described on n criteria. Typically, a MCDA problem faces a small number of alternatives (from

5 or 6 to less than 50). The number of criteria is also small ranging between 3 and about 10. It should be noticed that the real data can be considered as *example* data but not as *sampled* data as in the classical statistical sampling theory framework: the data set is not obtained by a random sampling, as the data has been generally previously selected for their interest. Therefore it is difficult to infer the distribution of each criteria from the data, as there exists a observation bias.

Since the margins are unknowns, it is preferable to use normalized ranked data to estimate the copula parameters. This avoids the problem of estimating the marginal distribution. However, we need to estimate these distributions in order to transform the simulated data (whose margins are uniformly distributed) into the original scale of the data. Two different solutions can be considered:

- choose a parametric form of distribution (Gaussian, uniform...) for the criteria and estimate its parameters, or
- use a non-parametric approach for the marginal distribution.

We chose to use the empirical distribution invert function which is a fully non-parametric approach. The inconvenience stands in the fact that we can only infer marginal distribution contained between the observed (real) minimum and maximum for each criterion. Therefore extrema values could be not so well simulated.

In order to avoid problems due to count data we assume that the margins are absolutely continuous. Thus, the representation in (2) is unique.

3.2 Simulation scheme

We use the statistical software R to perform the numerical experiments. The simulation process has been implemented in the `CDVine` package [9]. The input data set is a numeric performance matrix. To obtain a simulated dataset we follow these steps:

- Step 1.** Transform original data into copula data, i.e. purely ordinal distributions for each criterion.
- Step 2.** Select a C or a D vine structure via the function `CDVineCOPSelect` proposed in the package `CDVine`. Parameters of this function are the choice between C-Vine or D-Vine structure to be selected, and the selection criterion (AIC or BIC).
- Step 3.** Estimate the parameters of all the pair copula jointly through the maximization of the pseudo likelihood. This step is performed via the function `CDVineMLE` proposed in the package.
- Step 4.** Simulate the desired number of data via the function `CDVineSim` proposed in the package `CDVine`.
- Step 5.** Transform back copula data into real-like data via the inverse of the empirical cumulative function.

3.3 Evaluation

The testing step consists in the analysis of the differences between the set of real data and the set of simulated one. We want to detect if there is any difference between both sets and quantify the difference if any. An acceptable simulation procedure would yield on simulated data that is indistinguishable from the real data.

Since we are interested on a joint multivariate probability structure, using classical univariate tests (e.g. Kolmogorov-Smirnov test) on the margin of the joint distribution is clearly not sufficient. However, the simulation scheme must warranty that these margins are correctly simulated as well as the joint structure.

One could then rely on clustering methods to split the mixed datasets of real and simulated data into two clusters. Then, one computes a confusion matrix using the classes obtained from the clustering methods and the real labels (real vs simulated) and tests for independence through a χ^2 test. The k -means method is one of the most common and popular unsupervised clustering method. However, this method should be useless here, as it will always conclude to the confusion of real and simulated data as long as the marginal distributions will be close. This clustering method is able to capture clusters that are not in the same place in the possible data space, but is less able to capture clusters that have different structures in the same subspace.

Alternatively, one could use a binary classifier to test whether the merged data is easy to discriminate in terms of the added labels *real* vs. *simulated*. We use the Random Forest algorithm [10] as a supervised learning method. This algorithm allows to estimate the in-fit sample error rate, that is the proportion of alternatives that are wrongly classified. For this, the algorithm constructs many binary trees classifiers on bootstrapped subsets of the merged dataset. Then, it test the classifier on the remaining alternatives and computes the error rate. The quality indicator we look at is the mean global error ratio computed over all the classifiers constructed by the Random Forest algorithm. Heuristically, the higher the ratio the better it is in our case as it indicates that there is more and more confusion between real and simulated data.

A more formal way of measuring the quality of the simulation is to test the existence of differences between the simulated and real data. For instance, one could use a flexible multivariate ANOVA [26]. We do not explore this method in this paper.

3.4 Results

We tested the elicitation process on 3 data sets obtained from the MCDA data set repository [22]. Let us present the three selected cases.

- Case 1.** A data set of farms evaluated on the animal welfare, described in [6]. The dataset is composed of 29 farms described on 4 criteria valued between 0 and 100.
- Case 2.** A data set of 27 scenarios for radioactive waste management, with regard to 4 criteria, described in [11].
- Case 3.** A data set of 243 virtual peach ideotypes with regard to 3 criteria described in [17].

The three data sets are represented on the left panels of Figures 1, 2 and 3 respectively. These panels contain all the pairwise scatter plots for each data set on its upper triangle. On the lower triangles we represent each estimated pairwise copula density by means of contour plots. These pair copula are the elemental brick on the construction of the vines. It is possible to remark different kinds of probability structures and dependence between the three cases. For instance, the contour plots show spherical shapes in Figure 1 and elliptical shapes in Figure 2 which can be associated to multivariate normal or t distributions. The shapes of the contour plots in Figure 3 are more intricate and therefore represent more complex dependence structures.

In order to obtain a estimation of the quality of the simulation procedure we repeated the simulation scheme (see 3.2) 1000 times, producing then 1000 simulated data sets for each of the three real data sets. The simulated data sets have the same dimensions as the real data sets they are simulated from. Figures 1, 2 and 3 allow to visually inspect one of the replicates of the simulation procedure for each of the MCDA data sets. On the first two cases, it is hard to tell

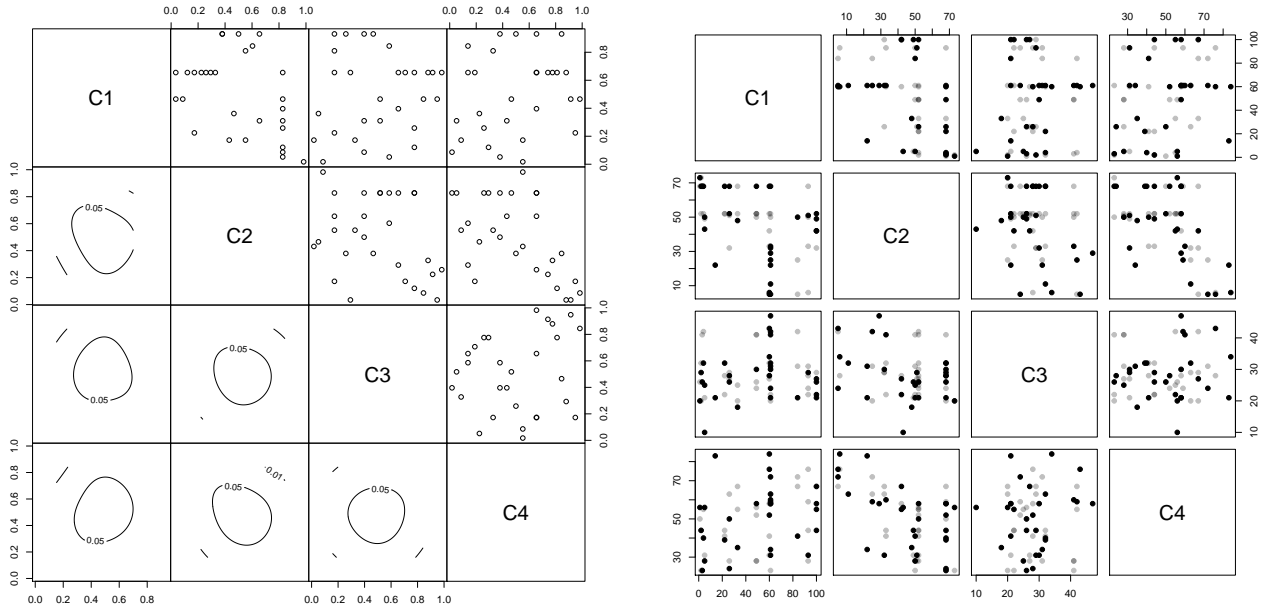


Figure 1: Animal welfare dataset [6]. On the right, the pairs plot of the data points (black dots) and simulated points (gray dots). On the left, the original data points are represented as normalized ranks on the upper triangle and the estimated pair copulas as shown as contour plots on the lower triangle.

that the global pattern of the data is not respected, however there are some simulated data points that lay in zone of low real data density. On the third case, it is more clear that the structure of the simulated dataset does not necessarily follow the structure of the real data. In particular, we see how simulated data points lay far away from the strong structure of real data set.

For each simulation we tested the univariate and multivariate adequacy of the simulated datasets to the real datasets. We used the Kolmogorov-Smirnov test on each margin and the proposed evaluation (see 3.3) using the Random Forest intrinsic error ratio. The Kolmogorov-Smirnov test is marginally rejected at the level of 5% (the maximum rejection was 5 times out of 1000 replication using the first dataset and the first variable) so we do not include the results here. The obtained average error ratios using Random Forest are presented in table 1 which is detailed and analysed in the next section.

In order to ensure that results are not purely due to randomness, we also produced 1000 simulated data sets without any hypothesis of dependence between criteria, i.e. we generated criteria values independently, following only the marginal distributions for each criterion on each dataset. The results of such simulation are also listed in table 1.

Data Set	RandomForest Error Ratio	
	With copulas	Without copulas
Case 1 [6]	0.534	0.5
Case 2 [11]	0.573	0.012
Case 3 [17]	0.204	0.119

Table 1: Average error ratio of the Random Forest classifier for each of the MCDA datasets for the simulation scheme using copula to model dependence and without any dependence structure.

A higher error ratio shows that it is more difficult to distinguish between learning data and simulated data with the use of copulas, and as a consequence we consider that the simulation is of better quality. If we consider that a higher error ratio implies a simulation of higher quality, then we notice that for all the proposed datasets the quality of the simulated data seems to be better when we use copulas than under the independence hypothesis. However, the differences among the three cases are not negligible. For example while in the first case both error rates are very close (0.534 under the dependence hypotheses against 0.500 under the independence hypothesis), in the second case the error rates are quite different (0.573 under the dependence hypothesis against 0.012 under the independence hypothesis). Let us examine more in detail these results.

First we used vine tree plots to graphically represent the estimated dependence structure. Vine tree plots represent dependent pair copulas as connected components on a graph. The non connected components are the (conditionally) independent couples of variables. When a pairwise dependence exists, the associated edge indicates the strength of the association and a label is placed together with the empirical Kendall's tau as well as the retained copula. Vine tree plots for our experiment are presented on Figure 4. With this representation it is easy to see the low dependence structure of the first case (where only one non independent pair copula is estimated), and also the strong structure observed in case 3 where all the possible pairwise components are linked together. Finally the case 2 is somehow more interesting because the dependence structure is present at some levels of the disaggregation and for some variables.

We now try to interpret the obtained results in terms of the practical problem associated to each one of the cases we studied.

Case 1. [6] presents data that are very weakly dependent (see figure 1). There, the representation of the data and the used copula seems to indicate that the dataset could be correctly represented using a spherical copula. Therefore, a simulation under the independence

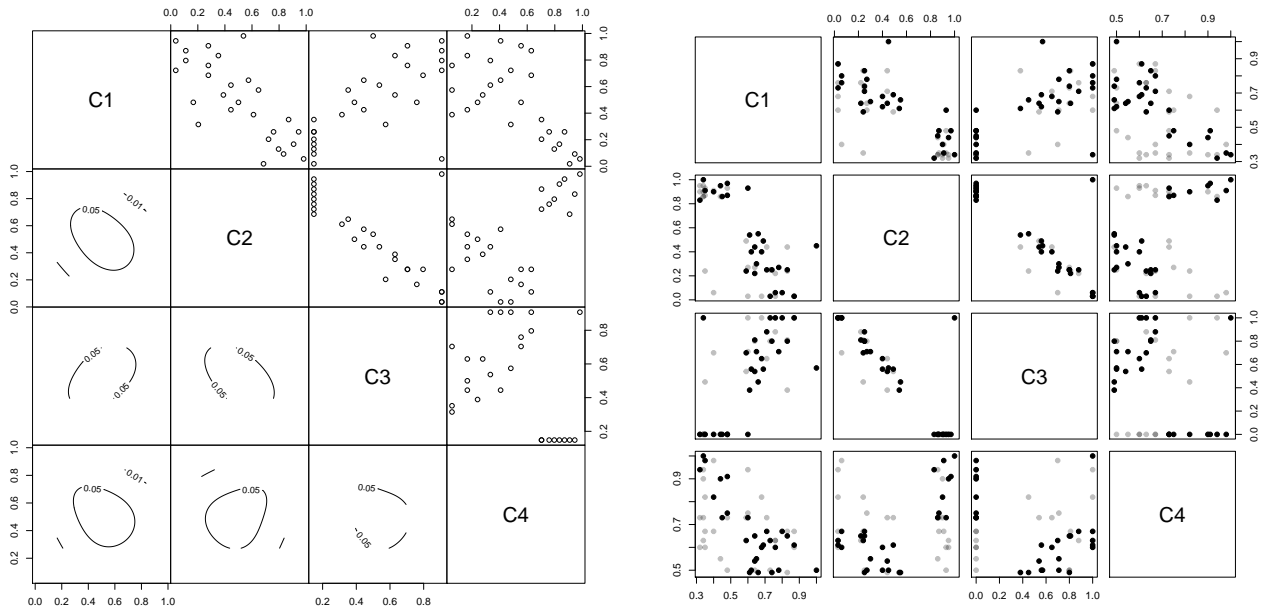


Figure 2: Radioactive waste management dataset. On the right, the pairs plot of the data points (black dots) and simulated points (gray dots). On the left, the original data points are represented as normalized ranks on the upper triangle and the estimated pair copulas as shown as contour plots on the lower triangle.

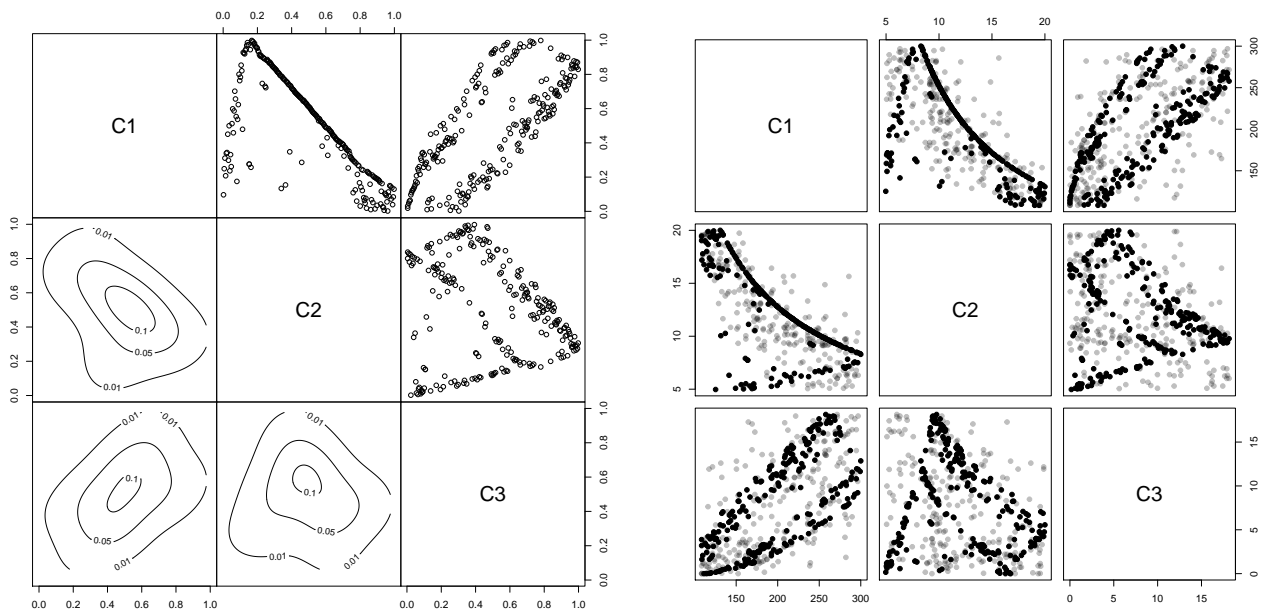


Figure 3: Peach idotypes dataset. On the right, the pairs plot of the data points (black dots) and simulated points (gray dots). On the left, the original data points are represented as normalized ranks on the upper triangle and the estimated pair copulas as shown as contour plots on the lower triangle.

hypothesis will give good results and will not be very improved by the use of copulas.

Case 2. [11] is a very interesting case as the data are more linked by a non-linear relation (see figure 2 for the representation of the data and the used copula). In this case the use of copulas permits to really improve the quality of the simulation by taking into account these links.

Case 3. [17] is a very special case since all the alternatives in the dataset are situated in a 3D-surface which is a Pareto front (see 3 for the representation of the data and the used copula). Simulating data without the constraint of being in the surface leads for sure to absurd solutions. The use of copulas in this situation can weakly improve the quality of the data, but cannot of course use the special surface structure of the data to better simulate new

alternatives. The simulation process should be linked with a cleaning phase where only pareto-optimal solutions should be kept in the dataset.

4 Conclusion

The objective of our work is to propose to the community a practical tool to simulate "real-like" data from a real dataset. We focused on the way to take into account weak and non-linear links between criteria and proposed a solution based on the use of copulas to model these links. We have shown that the use of copula increases the quality of the simulated data compared to the simple model only based on the use of the marginal distributions. The proposed process is based on an automatic learning of the copula model and parameters. However, we can imagine that the expert can define the used model of copula and/or part of the parameters if needed.

The use of copulas to simulate new MCDA data from a set of real ones seems to be validated by the tests we made. In each case data are of better quality with the use of copulas than if we simulate data under the hypothesis of complete independence between criteria. However, the use of copulas is of higher interest when the criteria are linked by a weak relation: if no relation exists between criteria one can simulate criteria values independently; if a strong (and hidden) relation exists between criteria copulas can fail at representing it.

Moreover, it should be noticed in a multicriteria combinatorial optimization point of view that we only generate alternatives with credible criteria values. We do not check if these alternatives correspond effectively to feasible solutions or not.

Perspectives of this work are the following:

- to provide an efficient similarity-index to test the similarity between real and simulated data;
- to develop an available R service for anyone to simulate MCDA data from a learning dataset;
- to study the effect of criteria number and alternatives number in the learning set on the quality of the simulation;
- to propose a process using copulas to simulate data directly from indications of the DM without any learning dataset.

REFERENCES

- [1] K. Aas, C. Czado, A. Frigessi, and H. Bakken. Pair-copula constructions of multiple dependence. *Insurance: Mathematics and Economics*, 44(2):182–198, 2009.
- [2] N. Ait-Taleb, V. Brison, and M. Pirlot. Generating multicriteria data similar to real data using Bayesian nets. In *11th Decision Deck Workshop*, 2013.
- [3] C.A. Bana e Costa, J.M. De Corte, and J.C. Vansnick. On the mathematical foundation of MACBETH. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*, pages 409–443. Springer Verlag, Boston, Dordrecht, London, 2005.
- [4] T. Bedford and R. M. Cooke. Vines—a new graphical model for dependent random variables. *Ann. Statist.*, 30:1031–1068, 2002.
- [5] R. Bisdorff, P. Meyer, and T. Veneziano. XMCD A : a standard XML encoding of MCDA data. In *EURO XXIII : European conference on Operational Research*, pages 53–53, 2009.
- [6] R. Botreau and A. Rolland. Evaluation multicritère du bien-être animal en ferme : une application des méthodes développées en aide à la décision multicritère. In *8ème congrès de la ROADEF, Clermont-Ferrand*, 2008.
- [7] D. Bouyssou, D. Dubois, M. Pirlot, and H. Prade, editors. *Concepts and Methods of Decision-Making*. Wiley-ISTE, 2009.
- [8] J.P. Brans and B. Mareschal. PROMETHEE methods. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*, pages 163–196. Springer Verlag, 2005.
- [9] E.C. Brechmann and U. Schepsmeier. Modeling dependence with c- and d-vine copulas: The r package cdvine. *Journal of Statistical Software*, 52(3):1–27, 2013.
- [10] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.
- [11] Th. Briggs, P.L. Kunsch, and B. Mareschal. Nuclear waste management: An application of the multicriteria {PROMETHEE} methods. *European Journal of Operational Research*, 44(1):1–10, 1990.
- [12] L. Dalla Valle. Official statistics data integration using copulas. *Quality Technology & Quantitative Management*, 11(1):111–131, 2014.
- [13] J. Figueira, V. Mousseau, and B. Roy. ELECTRE methods. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*, pages 133–53. Springer Verlag, Boston, Dordrecht, London, 2005.
- [14] M. Grabisch. The application of fuzzy integrals in multicriteria decision making. *European Journal of Operational Research*, 89:445–456, 1996.
- [15] R.L. Keeney and H. Raiffa. *Decisions with multiple objectives: Preferences and value tradeoffs*. J. Wiley, New York, 1976.
- [16] A. J. McNeil, R. Frey, and P. Embrechts. *Quantitative Risk Management*. Princeton Series in Finance, 2005.
- [17] M.-M. Memmah, A. Rolland, and B. Quilot-Turion. Multicriteria sorting methods to select virtual peach ideotypes. *International Journal of Multicriteria Decision Making*, to appear, 2014.
- [18] P. Meyer and S. Bigaret. Diviz: A software for modeling, processing and sharing algorithmic workflows in MCDA. *Intelligent decision technologies*, 6:283–296, 2012.
- [19] R.B. Nelsen. *An Introduction to Copulas*. Springer, second edition, 2006.
- [20] Z. Pawlak. Rough sets. *International Journal of Information and Computer Sciences*, 11:341–356, 1982.
- [21] Z. Pawlak. *Rough Sets. Theoretical Aspects of Reasoning about Data*. Kluwer, Dordrecht, 1991.
- [22] A. Rolland and T.M.T. Tran. A data set repository for benchmark in mcda. In *DA2PL 2014 Workshop From Multiple Criteria Decision Aid to Preference Learning*, page ??, 2014.
- [23] B. Roy. *Multicriteria Methodology for Decision Aiding*. Kluwer Academic, Dordrecht, 1996.
- [24] T.L. Saaty. *The analytic hierarchy process*. McGraw Hill International, New York, 1980.
- [25] A. Sklar. Fonctions de répartition à n dimensions et leur marges. *Publications de l'Institut de Statistique de l'Université de Paris*, 8:229–231, 1959.
- [26] M.A. Zapala and N.J. Schork. Multivariate regression analysis of distance matrices for testing associations between gene expression patterns and related variables. *Proceedings of the National Academy of Sciences*, 103:19430–19435, 2006.

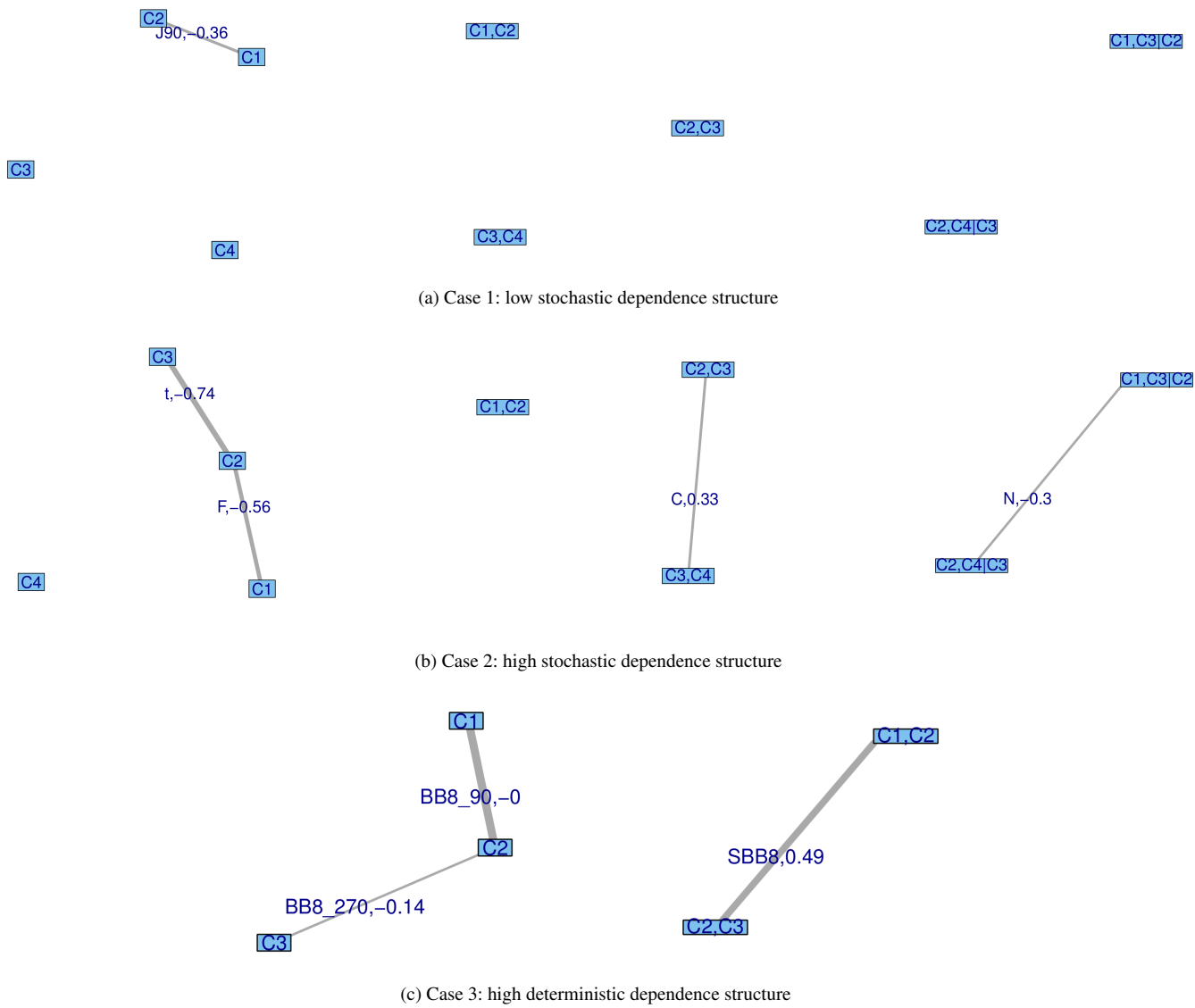


Figure 4: Tree vine plot for the three cases studied. Nodes represent the pair copula components. Edges are present where a significant dependence is estimated between pair copula.