European Journal of Decision Process manuscript No.
(will be inserted by the editor)

Simulation of Multicriteria Data

Jairo Cugliari · Antoine Rolland

Received: date / Accepted: date

Abstract For several reasons like benchmarking of MCDA methods, the MCDA community should be interested in the production of simulated multicriteria datasets based on real datatsets. Our goal in this paper is to propose several methods to simulate these new multicriteria data from an existing dataset. Simulated data should be as similar as possible to the initial dataset, including the capture of specific structure in the initial dataset (if any). We propose here to study independent sample, PCA-based sample and copula based sample and to determine which one best succeed in generating new data on demand. The copula based method seems the best one to reproduce specific links between criteria.

Keywords Simulation · MCDA Data · Copula

1 Introduction

As pointed out in [9] and [8], there is a need of simulated multicriteria datasets to be used by the Multicriteria Decision Aiding (MCDA) community . One of the reasons is that many different MCDA methods have been already proposed, each of them adapted to specific data or environment. Even if the axiomatic foundations of these methods have been generally well studied (see [6] for a first approach), it is often difficult to evaluate `in practice` between these MCDA methods (see [3], [6] or [15] for a survey). Therefore, testing these methods on several situations, using real datasets, should improve our understanding of advantages and inconveniences of each method. But these "real-world datasets" are generally not appropriate for such a task, for several reasons. In many cases, only very few data are available. From a preference learning point of view, the dataset may be so limited that it is too small to be divided into a test subset and a validation subset. Researchers should also desire to have more data with a specific shape to test the proposed methods onto data with plausible values on each criteria. Our goal is to propose one or several methods to simulate new multicriteria data from an existing dataset. Simulated data should be as similar as possible to the initial dataset. Therefore,

Universit de Lyon, Lyon 2, ERIC EA 3083 E-mail: name.surname@lyon2.fr

the proposed methods are supposed to be able to capture the specific dependence structure of the initial dataset (if any), and then generate new data on demand.

Good practices in MCDA point out that values on criteria should be as statistically independent as possible. It just means that the values taken on different criteria should not be functionally related (linearly or not) to avoid redundancy. Note that this statistical independence is not the preference independence between criteria (see [14]). Each time we mention independence in this paper we refer to statistical independence. But in real life the values taken by an alternative on different criteria are generally functionally related at some extent. Therefore, multicriteria data cannot be well simulated using only statistical independent sampling on each variable. The problem is then to model the dependence structure between criteria in a plausible way. While in some cases the functional link describing the dependence structure may be linear, it may be the case where a more flexible functional link is needed.

As introduced in [9], we propose to use a statistical approach to overcome this difficulty that can be summed up in two steps. First we borrow from other disciplines simulation schemes suitable for MCDA data. We use tools that range from quite classical methods like Principal Component Analysis (PCA) to modern simulation approaches based on copulas. Second we evaluate the quality of the simulated data by testing its similarity to the original non artificial data.

Our choices are guided by a increasing level of complexity on the multivariate dependence of the data. When this dependence can be correctly described by linear relationships we expect the PCA to detect them. Then, the uncorrelated linear factors obtained from PCA may be better simulated by independent draws on each factor. If the relationships between criteria were non linear, we propose to use copula which aims at model those interactions. Basically, a copula is a function that describes a multivariate distribution in terms 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 other work about the simulation of multicriteria data except a tentative using Bayesian network presented in [2].

In the present paper, we first propose in Section 2 to detail the proposed methods, including a fully independent simulation method, simulation method using Principal Component Analysis (PCA), and simulation method based on copulas. In Section 3, we propose a quality analysis of the simulation results regarding the three methods (independent sampling, PCA, copulas). Last in Section 4 we experiment these methods on real datasets and show that PCA and copulas-based simulators lead to simulated datasets of higher quality than simple independent samples.

2 Materials and methods

We describe in this section the three frameworks we use to create simulated MCDA data from available dataset that we describe by a matrix $\mathbf{X}$ with n rows and p columns. Each column represents a criterion (variable). Each row represents an alternative. The first method does not take into account any dependence structure

of $\mathbf{X}$ and so simulate values on each criterion independently from others. While this approach can be useful in some cases, in most real life applications the columns of $\mathbf{X}$ can not be considered totally independent. Therefore, we incorporate this element on the simulation scheme for the second and third methods.

2.1 Simulation by independent draws

The first framework uses the inverse of the classical probability integral transform, which is a classical simulation method on univariate distributions ([10]). That is, if a random variable, say Y, has a distribution function F_Y, then one can simulate realizations of Y in two steps. First, one draws a realization u from a standard uniform law (i.e. with support on the unit interval). Then, the realization y for Y is obtained as $y = F_Y^{-1}(u)$, where

$$F_y^{-1}(u) = \inf\{y : F_Y(u) \geq y\} \tag{1}$$

is a generalized inverse of F (since the distribution functions are weakly monotonic and right-continuous).

Our starting point is the set $y_1, \ldots, y_n$ of size n containing realisations of the target variable with an unknown distribution function. Therefore, we estimate F_Y by means of the empirical distribution function,

$$\hat{F}_Y(y) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{\{y_i \leq y\}}, \tag{2}$$

which is, for each y a simple average of indicators. Graphically, the empirical distribution function is a stepwise function with jumps at the observed realizations $y_1, \ldots, y_n$ an so inducing a restriction to the realization: only observed data points can be obtained with the simulation. In order to relax this constraint we apply an additional smoothing step to obtain from (2) a continuous version using interpolation splines. Then, one need to simulate standard uniform realizations u and apply (1) replacing F_Y by the smooth version of $\hat{F}_Y$. Since each column is treated independently, the simulation scheme for $\mathbf{X}$ follows the same steps column-wise.

2.2 Independent draws on latent factors

The independent draws approach is somewhat disappointing when applied to MCDA framework, because the eventual dependence structure between the alternatives is not taken in consideration. To overcome with this drawback one may consider a simple transformation of the dataset $\mathbf{X}$ to create latent factors over which the independent hypothesis will be more reasonable.

Principal Components Analysis is a very popular data analysis tool which allows one to construct a new coordinate system to better represent a multivariate dataset. The construction can be seen as a sequential search of a new variable (also known as factor) that is a linear combination of the original variables (a direction) in order to maximize the variance of the projected scatter the direction. Moreover, the directions are taken to be orthogonal to the precedent ones which gives a new

orthogonal coordinate system. In practice, all the directions are obtained simultaneously thanks to the singular value decomposition of a centred (and usually standardized) version of $\mathbf{X}$.

Actually, if the dependence structure of $\mathbf{X}$ is only linear, one may rely on the principal components analysis (PCA) to extract orthogonal variables which explain the best the variability on the data. Moreover, if $\mathbf{X}$ follows a multivariate Gaussian distribution this approach would generate independent factors. Our aim is not to push so far the independence hypothesis but to have a simple approach that will overcome the problems mentioned before.

Concretely, we decompose $\mathbf{X}$ using the classical principal component analysis to get a new matrix $\mathbf{F}$ with the same dimensions as $\mathbf{X}$. We keep the the barycentre and scale of the original dataset by an appropriate centring and standardization on the columns of $\mathbf{X}$. Then, we apply the independent draws approach on the columns of $\mathbf{F}$ in order to get a simulated set of latent factors. Using the well known reconstruction formula for PCA, the latent factors are used to get a simulated centred and standardized version of $\mathbf{X}$. Finally, the original barycentre and scales are incorporated to yield on the simulated version of $\mathbf{X}$.

Notice that while many matrix decomposition schemes exists, most of them can not be used because they lack off a reconstruction formula (i.e. Independent Components Analysis). The high computing performance of PCA gives an additional argument for our choice.

2.3 Copulas

Simple linear relationships may not suffice to describe the dependence structure of the criteria in MCDA. Then, copula construction may be promising alternative. In this section we recall some basic notions about modelling dependence with copulas (see [13] for a formal presentation of the subject).

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

$$C(u_1, \ldots, u_p) = P(U_1 \leq u_1, \ldots U_n \leq u_p) \tag{3}$$

A central result on copulas is the theorem introduced by [16] which allows one to represent any $p-$variate cumulative distribution function $F(x_1, \ldots, x_p)$ of the random vector $\mathbf{x} = (x_1, \ldots, x_p)$ as

$$F(x_1, \ldots, x_p) = C(F_1(x_1), \ldots, F_p(x_p)), \tag{4}$$

where $F_1(x_1), \ldots, F_p(x_p)$ 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 introduced in [13] which identifies the copula from the joint and marginal distribution

$$C(u_1, \ldots, u_p) = F(F_1^{-1}(x_1), \ldots, F_p^{-1}(x_p)). \tag{5}$$

Intuitively, the probabilistic structure of the random vector $\mathbf{x}$ is the result of coupling the marginal behaviour 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 $\mathbf{x}_1, \ldots, \mathbf{x}_n$ 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, see [13]). If F is absolutely continuous, then we use the chain rule to write the density equivalent to equation (4)

$$f(x_1, \ldots, x_p) = c(F_1(x_1), \ldots, F_p(x_p)) f_1(x_1) \ldots f_p(x_p) \tag{6}$$

where the copula density function c is given by

$$c(u_1, \ldots, u_p) = \frac{\partial^p C(u_1, \ldots, u_p)}{\partial u_1, \ldots, \partial u_p} \tag{7}$$

The difficulty of this problem depends on the data dimension p. In the bivariate case, e.g. $p = 2$, only one pair-copula must be estimated and many solutions have been already proposed to do so (see for example [12, Chapter 5]). However, several of these approaches are not feasible in higher dimension spaces.

To avoid some problems that arise on high dimension datasets, [4] propose a pair-copula construction (PCC) in order to decompose the multivariate joint density of $\mathbf{x}$ into a cascade of building blocks called pair-copula.

As before f is the joint density of $\mathbf{x}$ which is factorized (uniquely up to a relabelling of the elements of $\mathbf{x}$) as

$$f(x_1, \ldots, x_p) = f(x_p) f(x_{p-1}|x_p) \ldots f(x_1|x_2, \ldots, x_p). \tag{8}$$

Then, one can write each of the conditional densities on (8) using (6) recursively which yields on this general expression for a generic element x_i of $\mathbf{x}$ given a generic conditioning vector v

$$\begin{aligned} f(x_i|v) =& c_{x_i, v_j|v_{-j}} \left(F(x_i|v_{-j}), F(v_j|v_{-j})\right) \\ & \times f(x_i|v_{-j}). \end{aligned} \tag{9}$$

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.

Vines copulas have been proposed to classify alternatives factorization of (8) into a structured graphical model by [4]. This construction allows highly flexible decompositions of the (possibly high) dimensional distribution of $\mathbf{x}$ 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). We use the scheme called D-vines, where 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 those that were already modelled (see [1] for a more detailed exposition of this construction and other decomposition schemes).

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

$$h(x|v, \theta) = \frac{\partial^p C_{x, v_j|v_{-j}}(F(x|v_j), F(x|v_{-j}), |\theta)}{\partial F(v_j|v_{-j})}, \tag{10}$$

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 p uniformly distributed random variables $w_1, w_2, \ldots, w_p$. Then use the probability integral transform of the corresponding conditional distribution:

$$\begin{aligned} x_1 &= w_1, \\ x_2 &= F^{-1}(w_2|x_1), \\ x_3 &= F^{-1}(w_3|x_1, x_2), \\ &\ldots \\ x_p &= F^{-1}(w_{p-1}|x_1, \ldots, x_{-1}). \end{aligned}$$

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

3 Numerical experiments

We propose in this section to study the influence of some factors on the simulation quality. Several factors have been identified which can have an influence on the simulation quality, as listed in Table 1.

Concept	Notation
Size of the learning set	N
Number of criteria	P
Marginal distribution of each criterion	D
Intensity of the correlation between criteria	R^2
Linearity of the correlation between variables	L
Monotonicity of the functional link between variables	M

Table 1: List of influential factors on simulation quality

All the experiments are implemented in `R`. Code for the data generators and the simulation schemes are available in `github.com/cugliari/simuMCDA`.

3.1 Generating random learning sets

Let $F = \{N, P, D, R^2, L, M\}$ be the set of identified factors and $F_{n,p,d,r,l,m}$ be a situation where factor N is fixed to value n, factor P to value p and so on. In order to measure the influence of each factors we propose a experiment protocol as follows:

1. for each $F_{n,p,d,r,l,m}$, generate randomly 30 sets of initial data I_i via the use of a Cholevsky matrix (to control r) and post-treatment to obtain l,m,d as needed.
2. for each set I_i, do 30 data simulations using the three methods introduced in Section 2. For each simulation, compute the p-value of a goodness-of-fit multivariate test. We use the empirical goodness-of-fit test proposed by [17] — see in Section 3.2 for details.

3. save the distribution of the 900 p-values for each $F_{n,p,d,r,l,m}$.

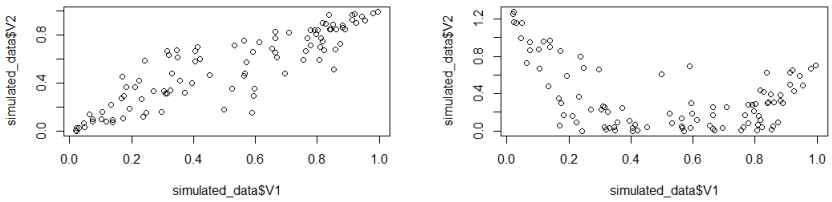
As developed below, we tested 5 different values for N, 10 different values for R, 3 different values for L, 2 different values for M and only one for D (uniform distribution) and for p ($p = 4$):

- Size of the learning set N: Intuitively, the bigger the learning set, the better the simulation will be. As our study particularly focus on small datasets,we decide to test our simulators for $N = 20, 30, 40, 50$ and 100 data in the learning set.
- Number of criteria P: the more criteria are used, the more difficult it will be to capture the links between all the different pairs (or subset) of criteria. A previous study presented by [8] showed that for a number of criteria between 3 and 6 there is no significant difference between the simulated data quality. On another hand, the tests we made with $P = 6$ showed that the computing time greatly increases with P. Therefore we decide to focus only on the case $P = 4$. Examples with more than 4 criteria are presented in section 4.1.2.
- The marginal distributions on the criteria: the use of the Cholevsky matrix to generate a dataset with a controlled determination degree leads to normal marginal distributions for the criteria. A post-treatment based on the rank of each criterion value turned the marginal distribution of each criterion into an uniform marginal distribution.
- Intensity of the correlation between criteria R^2: we measure the intensity of the correlation through the determination coefficient R^2 between two criteria. We test our simulators with R^2 varying from 0.5 to 0.95 by step of 0.05.
- The linearity of the correlation between variables L: as said above, simple linear relationships may not suffice to describe the dependence structure of the alternatives in MCDA. In order to represent functional but not linear dependencies between criteria, we make a post-treatment on 2 of the criteria. For each initial randomly generated set, we apply exponential function on criterion #3 and logarithm function on criterion #4. We therefore have in the same dataset linear (between criteria #1 and #2), exponential (between criteria #1 and #3) and logarithmic (between criteria #1 and #4) dependencies.
- The monotonicity of the correlation between variables indicates that on some situations MCDA data are not linked by a monotonic function. We simulate this by applying a post-treatment on the criterion #2 which turns the monotonic correlation of criteria #1 and #2 ("cigar" shape) into a non-monotonic correlation ("banana" shape). Examples of such simulations are shown in Figures 1a and 1b.

An example of a simulated learning set is shown in Figure 2.

3.2 Quality index

We state that data are correctly simulated if it is not possible to distinguish the real data and the simulated ones. So we need a tool that is able to distinguish two different distributions, such as a statistical multivariate goodness-of-fit test. But as pointed out by [11], non-parametric goodness-of-fit test that can be used in practice is something very hard to find. We then choose to use the goodness-of-fit



(a) Data simulation with cigar shape (b) Data simulation with banana shape

Fig. 1: Example of “banana” and “cigar” data set

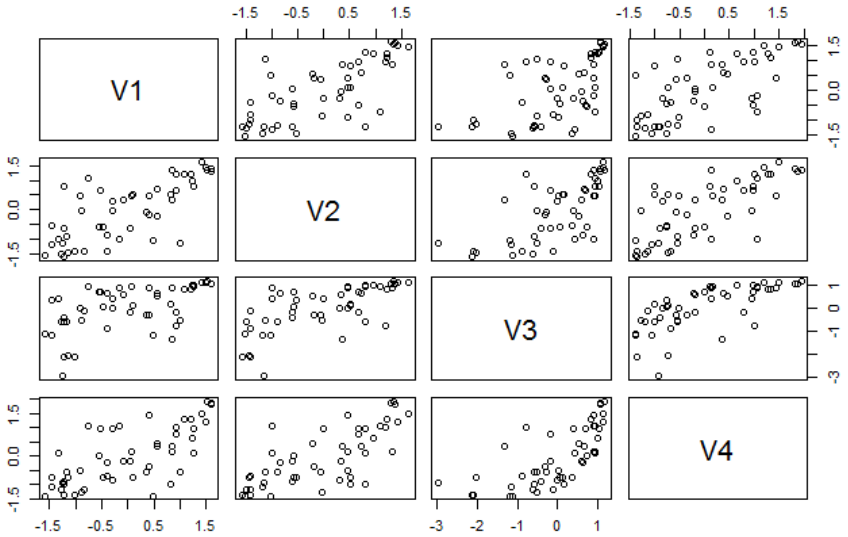


Fig. 2: Example of a simulated learning set

test proposed by [17], based on a geometric approach, and implemented in R. The null hypothesis H_0 is “the two multivariate distributions are the same”, versus hypothesis H_1 “the two multivariate distributions are different”. The test returns a p-value: if this p-value is less than a fixed threshold, then the difference between the two distribution is said to be statistically significant, and then the simulated data cannot be considered to have the same distribution as the real ones. The p-value can be seen as a quality index: the greater the p-value, the greater the simulation quality. Of course the p-value will change for each simulated data set. Therefore, as presented in Section 3.1, we study the result of 900 simulations via the boxplot of all the p-values. It is important to notice that the geometrical approach

on which the test is based supposes that all the variables are homogeneous. On the contrary, the geometrical distance has no sense.

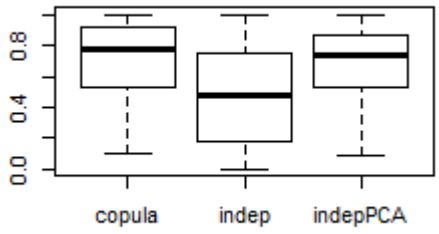


Fig. 3: Example of a quality boxplot

On Figure 3, one can observe that real and simulated data can easily be confused, as more than 75% of the simulated data sets have a p-value greater than 0.5 for the confusion test (remember that generally the threshold to reject H_0, equal distribution hypothesis, is a p-value less than 0.05). Other comparison processes have been studied. Especially, we tried to use supervised and unsupervised classification methods to determine whether real and simulated data can be distinguished using machine learning. However, these methods (SVM, k-means, random forest) did not managed to separate real and simulated data when the data have the same margin distribution, which is always the case here by construction.

3.3 Factors influence analysis

We present in this section an analysis of the influence of each identified factor in Section 3.1. Each analysis is based on the boxplot of the p-values of the goodness-to-fit test. Remember that the higher the p-values, the better the simulation quality.

3.3.1 Size of the learning set N

In figure **??** is shown the influence of the learning set size (the number of data in the learning set) on the quality of the simulation. One can see that for the independence method the quality of the simulation greatly decrease with respect to the learning set size. For PCA-based method the decreasing is less important. And a quasi-stability is observed for copula-based method which appears to be non-dependent to the learning set size.

3.3.2 Intensity of the correlation between criteria R^2

In figure 4 is shown the influence of the correlation intensity on the quality of the simulation. One can see that for all the three methods the quality of the

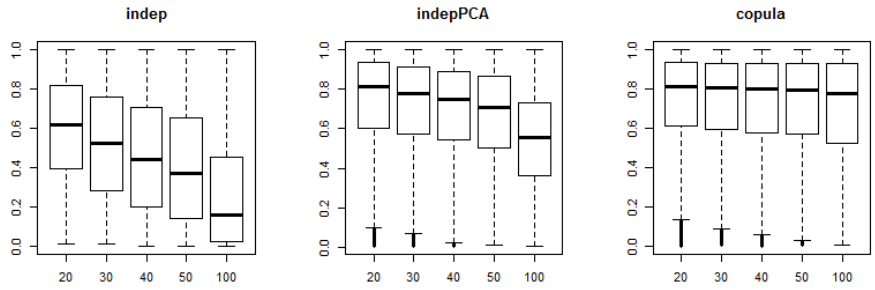


Fig. 4: Learning set size influence

simulation decreases with the increase of the correlation degree. The decrease is big in the case of the independent method, and less in the case of the PCA based and copula based methods. However this analysis should go deeper when linked with the dataset shape (banana or cigar).

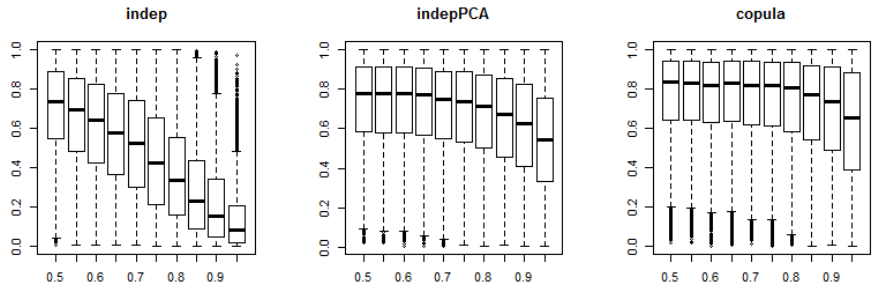


Fig. 5: Correlation intensity influence

3.3.3 Monotonicity of the correlation between variables M

As shown in Figure **??**, the impact of dataset shape is different for each method. For the independence method, the simulation quality is generally bad, but is a little bit better in the case of a "banana" dataset than a "cigar" dataset. It is the contrary for PCA and copula based methods, where simulations are globally good, and better in case of "cigar" shape than "banana" shape. We can see that when the correlation is strong with "banana" shape, both PCA and copula methods hardly detect the specific shape, and consequently are unable to well simulate these dataset. On Figure 5 we can see that the size has also a joint effect with the shape of the initial dataset. For a correlation intensity fixed at 0.9, the learning set size has clearly an influence on the ability of the copula method to well simulate

new data. When the learning set is too small with a "banana" shape, the dataset is not so different than a non-structured dataset and even the independent method produces simulations not so different than the initial dataset. When the structure is strong (100 items in the initial dataset and correlation index at 0.9), it is obvious that even the copula method simulates the specific structure of the dataset with difficulties.

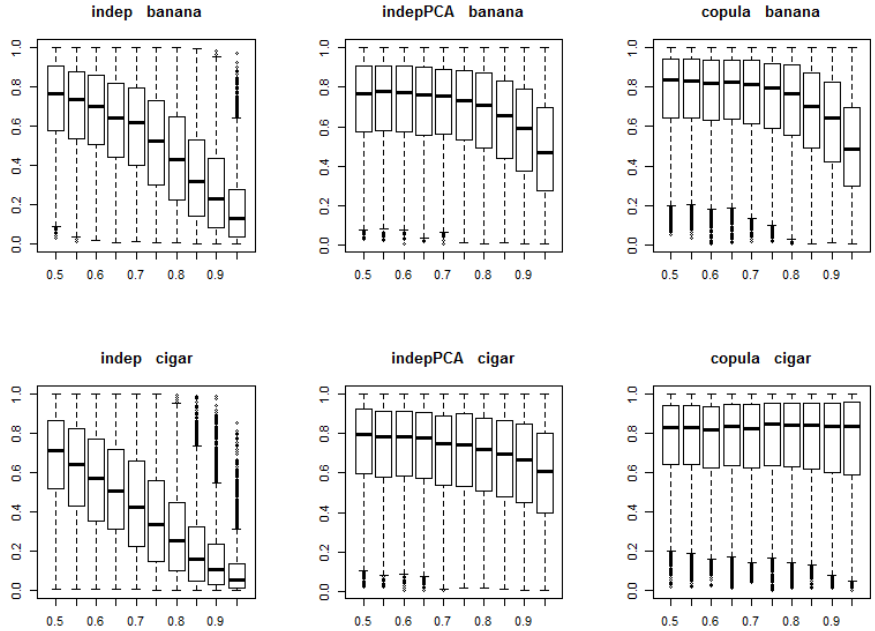


Fig. 6: Shape and correlation intensity influence

3.4 General conclusion: copula wins

As stated above, we can say that the copula method is better than the PCA method, which is better than the independent method. Results are better for "cigar" shape than for "banana" shape, and for "cigar" shape it is better to have more data in the learning set. The results of this small study should encourage all the practitioners to use a more accurate simulation process than just independent method. However, the computing time needed to use copula method can be a difficulty as it is significantly greater than the computing time needed by the two other methods. An example presented in Table 2 using data from the dataset detailed in Section 4.1.2 shows how the computing time increases with the number of criteria, for 88 alternatives.

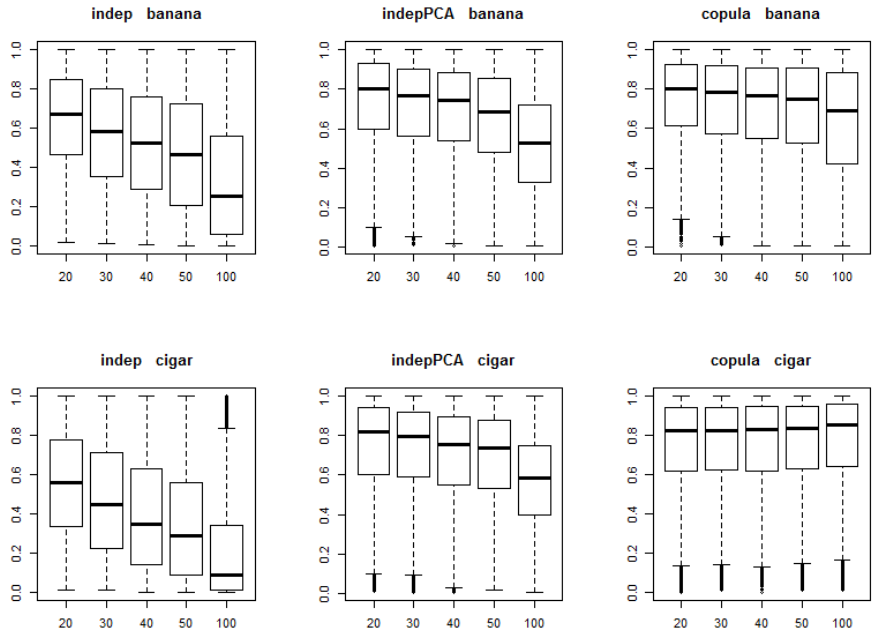


Fig. 7: Shape and data number influence

Nb. of criteria (p)	`indep`	`indepPCA`	`copula`
4	0.33	0.41	3.64
5	0.38	0.50	9.59
6	0.45	0.64	16.83
8	0.64	0.81	81.86
10	0.63	0.99	233.82
12	0.69	1.21	352.43
15	0.74	1.21	570.22
19	0.81	1.31	915.76

Table 2: Average computing time (in seconds) of the different simulation methods for 88 individuals (n) and criteria number (p) from 4 to 19.

4 Application on real datasets

4.1 Multicriteria dataset

4.1.1 Simulation on small datasets

We propose in this section to test the three different methods on some real datasets from the literature. The first one, proposed in [5], has 29 observations and 4 variables. The variables are almost independent and have the same scale, between 0 and 100. The second one was proposed in [7] and has 27 observations for 4

variables. The correlation coefficient between variables are around 0.7 and the variables have the same scale, between 0 and 100. For each dataset, we produced 1000 simulated datasets and then we draw the p-value boxplot as before. Results are shown in Figures 6.

As a result, we can observe that the PCA sample method seems to produce more accurate simulated data, even if the copula sample method leads also to good quality simulated data. For the first dataset, the three different methods are similar, even if the independent sample method does not produce the same quality datasets as the two others. It seems that even if no correlation is observed between the variables in the initial dataset, PCA and copulas sample methods are able to catch a small dependent link and therefore lead to more accurate data generation. For the second data set, it is very clear that the independent sampling method is not efficient: this case shows that most of the sampling produced by the independent methods can be distinguished from the initial dataset, whereas those produced with the copula sampling method, or even better with the PCA sampling method, can be considered as similar to the initial dataset.

These two examples give a good illustration that taking into account dependencies between variables (even if there are not obvious) leads to better simulated data than just independent sampling method. However, it is a surprise for us that on these examples PCA sampling method seems to produce better results than copula-based method.

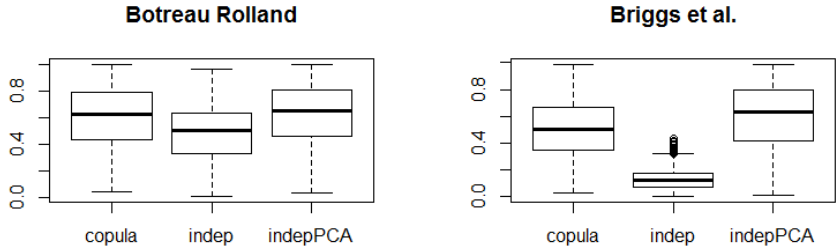


Fig. 8: Two real datasets

4.1.2 Simulation of french departments

France is administratively and geographically divided into about 100 different departments; The national statistical institute INSEE published on december 2016 a sustainable development-oriented portrait of the departments including 19 criteria[1]. A quick analysis shows that departments around Paris have a very specific profile. We then focused on the 88 other departments. Our question is "can we simulate new departments which have the same profile as existing one"? We present

[1] see `https://www.insee.fr/fr/statistiques/2512993`

in the following two different simulations with different numbers of criteria. As a set with 19 criteria needs a very long computing time to be simulated, especially with the copula methods, we propose to first compute simulations with a subset of 8 criteria having an average determination coefficient of 0.23 between each others has been computed (300 simulations). The results are shown in Figure 7. It is obvious that both PCA and copula based methods lead to good simulations whereas independent sampling leads to bad simulated sets. Moreover, the copula method gives information about the structure of the initial dataset. For example with the departments dataset on 8 criteria, the vine-copulas say that criteria C1 and C2 are linked by a Gaussian copula; this is coherent with the signification of C1 and C2, respectively life expectancy for men and women, which are strongly correlated. The vine-copula also says that C5 is linked with C4 through a Student copula, and with C6 with a Gumbel copula; this is also coherent as C4 is the poverty ratio, C5 the percentage of youth without employment, and C6 the part of population far from the basic health services, all criteria which characterize rural areas. As a third example, C7 is not copula-linked to other criteria which is also coherent as C7 represents the percentage of flood-prone area, which have no link with the other criteria. Therefore, PCA and copula based methods appear to succeed in modelling the correlation between criteria in a plausible way. An example of simulated department dataset with PCA and copula methods is shown in Figure 8. Please notice that, as pointed out in Section 3.2 the goodness-of-fit test is applied on scaled data, as the 19 criteria have very different scales, from a proportion between 0 and 1 to a number of inhabitants in millions.

The complete departments dataset includes 19 criteria. Most of the pairs of two criteria have no specific correlation. Only 18 pairs have a determination coefficient greater than 0.3, and only 6 greater than 0.5. We also tried to compute 300 new datasets simulations. The results are shown in figure 7. Again, PCA and copula based methods appear to succeed in modelling the data structure and producing new and plausible departments.

5 Discussion

We explored three strategies to simulate MCDA datasets from existing ones. Using artificially generated datasets to simulate new ones we have showed that:

- while computationally the fastest, statistically independent criteria simulation (`indep`) is in general a very poor strategy when used alone;
- statistically independent draws may be used after preprocessing the original dataset to obtain uncorrelated factors, for example with PCA;
- PCA preprocessing may distortion the dependence structure if it cannot be properly described by linear relationships, then the copula based strategy outperforms `indepPCA`
- the copula based simulation strategy is the slowest one and may be computationally prohibitive if the number of criteria is too large;
- on moderate and large dataset sizes with a not too big number of criteria, copula based simulation success to better capture the dependence structure.

In view of these conclusions we suggest the following practical considerations when simulate MCDA datasets:

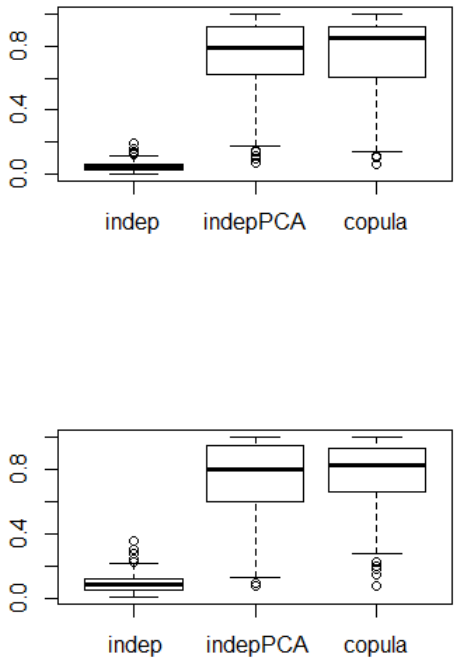
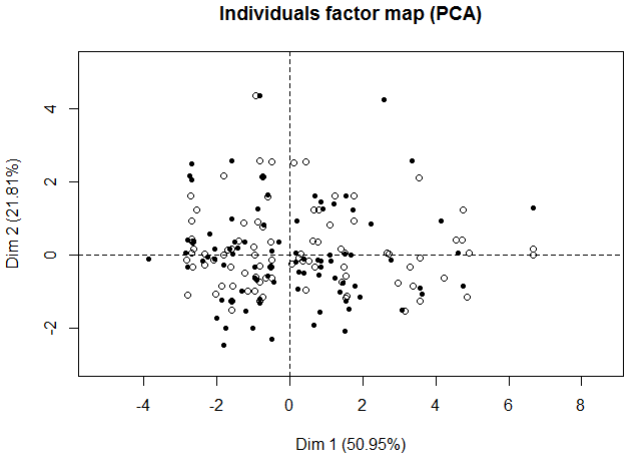


Fig. 9: Departments: first figure with 8 criteria, second one with 19 criteria

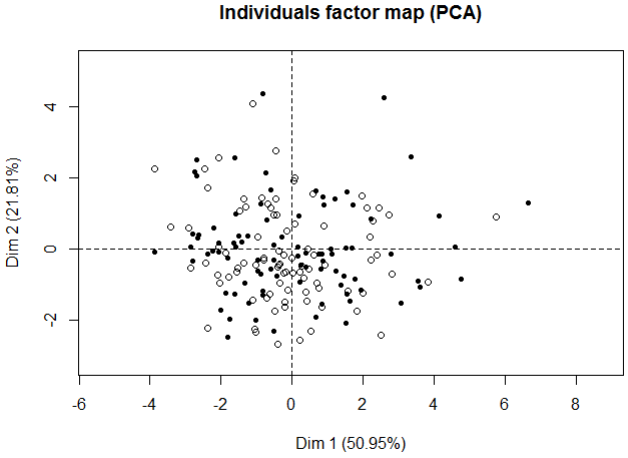
1. avoid using independent draws on criteria;
2. if the dependence structure is mostly linear, use PCA based simulation. Assessing whether linear dependence is a reasonable hypothesis may be done with the inspection of the singular values of the spectral decomposition of $\mathbf{X}$. If they rapidly decay to zero then factors will capture most of the richness of $\mathbf{X}$;
3. for non linear dependence structure use copula based simulation. However, attention must be paid to the number of alternatives (n should be large enough, say at least 20) and the number of criteria (p should be not too large, say below 20). Since this setting of non linear dependence structure, relatively moderate number of alternatives and relatively small number of criteria is the most common in MCDA, the copula based simulation strategy should be frequently used.

References

1. Aas, K., Czado, C., Frigessi, A., Bakken, H.: Pair-copula constructions of multiple dependence. Insurance: Mathematics and Economics **44**(2), 182 – 198 (2009). DOI



(a) Department simulation — PCA based method



(b) Department simulation — copula based method

Fig. 10: Department simulation — PCA projection — initial dataset is in filled dots, simulated dataset in circles with 8 criteria

http://dx.doi.org/10.1016/j.insmatheco.2007.02.001. URL `http://www.sciencedirect.com/science/article/pii/S0167668707000194`

2. Ait-Taleb, N., Brison, V., Pirlot, M.: Generating multicriteria data similar to real data using Bayesian nets. In: 11th Decision Deck Workshop (2013)
3. Bana e Costa, C., De Corte, J., Vansnick, J.: On the mathematical foundation of MACBETH. In: J. Figueira, S. Greco, M. Ehrgott (eds.) Multiple Criteria Decision Analysis: State of the Art Surveys, pp. 409–443. Springer Verlag, Boston, Dordrecht, London (2005)
4. Bedford, T., Cooke, R.M.: Vines–a new graphical model for dependent random variables. Ann. Statist. **30**, 1031–1068 (2002)

5. Botreau, R., Rolland, A.: 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)
6. Bouyssou, D., Dubois, D., Pirlot, M., Prade, H. (eds.): Concepts and Methods of Decision-Making. Wiley-ISTE (2009)
7. Briggs, T., Kunsch, P., Mareschal, B.: Nuclear waste management: An application of the multicriteria {PROMETHEE} methods. European Journal of Operational Research **44**(1), 1 – 10 (1990). DOI 10.1016/0377-2217(90)90308-X. URL `http://www.sciencedirect.com/science/article/pii/037722179090308X`
8. Cugliari, J., Rolland, A.: Simulating new multicriteria data from a given data set. In: DA2PL 2016 Workshop From Multiple Criteria Decision Aid to Preference Learning (2016)
9. Cugliari, J., Rolland, A., Tran, T.: On the use of copulas to simulate multicriteria data'. In: DA2PL 2014 Workshop From Multiple Criteria Decision Aid to Preference Learning, pp. 3–9 (2014)
10. Devroye, L.: Non-Uniform Random Variate Generation(originally published with. Springer-Verlag (1986)
11. McAssey, M.P.: An empirical goodness-of-fit test for multivariate distributions. Journal of Applied Statistics **5**(40), 1120–1131 (2013)
12. McNeil, A.J., Frey, R., Embrechts, P.: Quantitavie Risk Management. Princeton Series in Finance (2005)
13. Nelsen, R.: An Introduction to Copulas, second edn. Springer (2006)
14. Roy, B.: Multicriteria Methodology for Decision Aiding. Kluwer Academic, Dordrecht (1996)
15. Roy, B., Słowiński, R.: Questions guiding the choice of a multicriteria decision aiding method. EURO Journal on Decision Processes **1**(1), 69–97 (2013)
16. Sklar, A.: Fonctions de répartition à n dimensions et leur marges. Publications de l'Instiutut de Statistique de l'Université de Paris **8**, 229–231 (1959)
17. Székely, G.J., Rizzo, M.L.: Testing for equal distributions in high dimension. InterStat (5) (2004)