PSYCHOMETRIKA—VOL. 84, NO. 4, 941–985
DECEMBER 2019
https://doi.org/10.1007/s11336-019-09686-1

ROOTCLUS: SEARCHING FOR "ROOT CLUSTERS" IN THREE-WAY PROXIMITY DATA

LAURA BOCCI

SAPIENZA UNIVERSITY OF ROME

DONATELLA VICARI

SAPIENZA UNIVERSITY OF ROME

In the context of three-way proximity data, an INDCLUS-type model is presented to address the issue of subject heterogeneity regarding the perception of object pairwise similarity. A model, termed ROOTCLUS, is presented that allows for the detection of a subset of objects whose similarities are described in terms of non-overlapping clusters (ROOT CLUSters) common across all subjects. For the other objects, Individual partitions, which are subject specific, are allowed where clusters are linked one-to-one to the Root clusters. A sound ALS-type algorithm to fit the model to data is presented. The novel method is evaluated in an extensive simulation study and illustrated with empirical data sets.

Key words: clustering, INDCLUS, individual partitions, three-way proximity data.

1. Introduction

In many research domains (psychometry, sociometry, marketing), it can be of interest to investigate how the perception or evaluation of the similarities between objects (products or stimuli) may differ across several subjects (sources, experimental conditions, situations, scenarios or times).

Our concern here is referred to three-way two-mode data arranged as a set of square symmetric matrices of pairwise proximities between N objects provided by H subjects.

Clustering three-way similarity data is a complex and challenging task since proximity matrices actually subsume different classifications of the objects due to the subject heterogeneity.

Different approaches have been introduced to handle and account for the subject heterogeneity: The three-way clustering should represent a consensus of the individual object classifications, but such a consensus is actually representative only when no differences due to the subject heterogeneity occur and such an assumption is often unrealistic.

In order to account for the subject heterogeneity of three-way similarity data, Carroll and Arabie (1983) introduce the INDCLUS (INdividual Differences CLUStering) model as a three-way generalization of ADCLUS (ADditive CLUStering), originally formulated by Shepard and Arabie (1979) for two-way similarity data. ADCLUS assumes that objects are grouped into overlapping clusters and the similarity between objects equals the sum of the weights of all the clusters they belong to. Mirkin (1987) discusses the use of Qualitative Factor Analysis (QFA) methods for fitting ADCLUS in the special case where the qualitative factors are clusters. In the three-way framework, INDCLUS allows for extracting overlapping clusters of N objects from the proximities provided by H subjects which are supposed to differently weigh each cluster.

Electronic supplementary material The online version of this article (https://doi.org/10.1007/s11336-019-09686-1) contains supplementary material, which is available to authorized users.

Correspondence should be made to Donatella Vicari, Department of Statistical Sciences, Sapienza University of Rome, Rome, Italy. Email: donatella.vicari@uniroma1.it

Therefore, INDCLUS deals with the subject heterogeneity by assuming that even though all subjects share the same clustering of objects, they assign different patterns of weights to clusters so that the effect of each cluster on the proximities is subject specific.

Several extensions of INDCLUS have been proposed which present more flexibility in accounting for the subject heterogeneity.

Giordani and Kiers (2012) provide a fuzzified version of the INDCLUS model (FINDCLUS) which allows objects to belong to a partition with a fuzzy membership degree which is common to all subjects.

In the context of clustering and multidimensional scaling for three-way similarity data, the CLUSCALE (simultaneous CLUstering and SCAL[E]ing) model (Chaturvedi and Carroll 2006) combines additively INDCLUS and INDSCAL (Carroll and Chang 1970) searching for a common both (discrete) clustering representation and continuous spatial configuration of the objects. Thus, what is not accounted for by the INDCLUS term is handled by the INDSCAL part.

As a further extension of the INDCLUS model, Bocci and Vicari (2017) propose GINDCLUS, a Generalized INDCLUS model where the subject heterogeneity is accounted for by determining classes of subjects in addition to the common classification of the objects. Furthermore, the availability of possible information which is external to the three-way data is exploited to better account for the subject heterogeneity and object clustering.

Several other methodologies searching for simultaneous classification of subjects and objects had been before proposed, aiming at accounting for the subject heterogeneity of three-way proximity data.

In a least-squares approach, POP (Partitions Of Partitions) and PARSCLA (PARtition and least-Squares Consensus cLassifications Analysis) have been proposed by Gordon and Vichi (1998) and Vichi (1999), respectively, by searching classes of subjects where a single consensus partition of objects is found, while Vicari and Vichi (2009) fit several classification structures within each class of subjects.

In a maximum likelihood framework, Wedel and DeSarbo (1998) and Bocci, Vicari and Vichi (2006) fit finite mixture models which deal with the three-way heterogeneity by identifying unobserved classes of subjects each having a different classification structure of the objects.

In this paper, we do not deal with clustering of subjects, but in order to extract more information from the three-way proximities by accounting for and taking advantage of the subject heterogeneity, we move on from the INDCLUS model by considering that a unique common object clustering is not enough and that each subject may only *partially* share a common partition of some objects but differ in how some other objects are classified.

Specifically, the approach proposed here assumes that the N objects can be divided into two subsets: one including N_P objects, which subsume an underlying common perceptual structure of the pairwise proximities, while the other contains the leftover N_M objects whose proximities are differently evaluated by each subject.

Hence, the ROOTCLUS (ROOT CLUStering) model proposed here aims at jointly identifying:

(A) a *ROOT* partition of the N_P objects (i.e., a common partition) into non-overlapping clusters denoted *ROOT* clusters;
(B) H different subject-specific partitions of the remaining N_M objects into clusters linked one-to-one to the common ones. From now on, such subject-specific partitions are termed *Individual* partitions because they are different and peculiar for each subject.

The idea is that some common object clusters exist which account for the homogeneity of the subjects and form the roots at which the individual clusters intersect.

The model is formalized in a least-squares framework and an appropriate Alternating Least-Squares-type algorithm is given.

The paper is organized as follows. In Sect. 2, a real motivating example is presented to give an intuition of the problem we are dealing with. In Sect. 3, the ROOTCLUS model is formalized in a general framework and an appropriate algorithm is proposed in Sect. 4. An extensive simulation study and applications to real data are presented in Sects. 5 and 6, respectively. Finally, the last section is devoted to some concluding remarks.

2. Motivating Example

In order to give a flavor of the situation where the ROOTCLUS model could fruitfully provide a useful insight into clustering heterogeneous proximity data among subjects, we present an example of a real application, before discussing in detail the statistical framework of the model.

The data considered here refer to the classification of $N = 14$ sports (Volleyball, Horse-riding, Cycling, Athletics, Water Polo, Rugby, Martial Arts, Ski, Soccer, Fencing, Basketball, Swimming, Artistic Gymnastic, Tennis) provided by a number of college students in December 2017 at the Department of Statistical Sciences—Sapienza University of Rome (Italy). Specifically, the students were asked to sort the sports (given in random order) into three non-overlapping clusters on the basis of some common aspect not to be explicitly expressed. For each student, the partition provided was converted into a membership matrix and used to build a 14×14 similarity matrix for each student having ones only for the pairs of sports put together into the same cluster and zeros otherwise.

In order to study the gender differences in evaluating the sports, two matrices have been built from the similarity matrices of the students by summing up separately the single matrices of Males and Females and dividing them by their number (8 Males and 5 Females, respectively), so that, conditionally on gender, the percentage of times in which two sports have been assigned to the same cluster was assumed as a measure of their similarity (Table 1).

In this way, Males and Females are regarded as two "macro-subjects," representative of the two genders.

By a first inspection, it is clear that a different pattern in evaluating the similarity among sports is present and both matrices exhibit a strong underlying structure in clusters.

Moreover, in the clustering process such a structure is differently exhibited by Males and Females and some points can be highlighted about the heterogeneity of their similarities:

1. agreement for some sports generally allocated to the same cluster regardless of the gender (as for example Volleyball, Water Polo, Rugby, Basketball and Soccer which all Males and almost all Females assign to a cluster clearly identified as "team sports using a ball");
2. some sports quite clearly assigned to a cluster of similar sports by one gender but less clearly evaluated by the other gender: For example, Tennis is added to the "ball sports" and never classified with Gym and Athletics by Males, while Females put Tennis with the "ball sports" as many times as with Gym and Athletics;
3. sports allocated to a cluster to some (more or less) limited extent according to possible gender preferences (as Martial Arts).

From such considerations, we may deduce that Males and Females share their evaluations for some common clusters of sports, but they also differ in evaluating some other sports, so that one single partition is not enough to explain all the differences.

The idea underlying the proposed methodology is to find jointly A) common "root" clusters including only a subset of sports which is subsumed by Males and Females and, in addition, B) different gender-specific clusters of the remaining sports which are linked one-to-one to the roots to account for the gender heterogeneity.

TABLE 1.
Sport data by gender

	Volleyball	Horse-riding	Cycling	Athletics	Water Polo	Rugby	Martial Arts	Ski	Soccer	Fencing	Basketball	Swimming	Artistic Gym	Tennis
							Males							
Volleyball	100	0	0	0	100	100	0	0	100	12.5	100	12.5	0	62.5
Horse-riding	0	100	75	25	0	0	12.5	87.5	0	37.5	0	37.5	12.5	25
Cycling	0	75	100	25	0	0	0	87.5	0	37.5	0	50	25	12.5
Athletics	0	25	25	100	0	0	50	12.5	0	12.5	0	50	87.5	0
Water Polo	100	0	0	0	100	100	0	0	100	12.5	100	12.5	0	62.5
Rugby	100	0	0	0	100	100	0	0	100	12.5	100	12.5	0	62.5
Martial Arts	0	12.5	0	50	0	0	100	12.5	0	50	0	37.5	62.5	25
Ski	0	87.5	87.5	12.5	0	0	12.5	100	0	50	0	37.5	12.5	25
Soccer	100	0	0	0	100	100	0	0	100	12.5	100	12.5	0	62.5
Fencing	12.5	37.5	37.5	12.5	12.5	12.5	50	50	12.5	100	12.5	12.5	12.5	50
Basketball	100	0	0	0	100	100	0	0	100	12.5	100	12.5	0	62.5
Swimming	12.5	37.5	50	50	12.5	12.5	37.5	37.5	12.5	12.5	12.5	100	62.5	0
Artistic Gym	0	12.5	25	87.5	0	0	62.5	12.5	0	12.5	0	62.5	100	0
Tennis	62.5	25	12.5	0	62.5	62.5	25	25	62.5	50	62.5	0	0	100

TABLE 1.
continued

	Volleyball	Horse-riding	Cycling	Athletics	Water Polo	Rugby	Martial Arts	Ski	Soccer	Fencing	Basketball	Swimming	Artistic Gym	Tennis
							Females							
Volleyball	100	0	0	0	100	100	0	0	100	0	80	0	0	40
Horse-riding	0	100	80	0	0	0	60	80	0	80	20	40	0	20
Cycling	0	80	100	20	0	0	40	100	0	60	20	60	0	0
Athletics	0	0	20	100	0	0	40	20	0	20	0	60	80	40
Water Polo	100	0	0	0	100	100	0	0	100	0	80	0	0	40
Rugby	100	0	0	0	100	100	0	0	100	0	80	0	0	40
Martial Arts	0	60	40	40	0	0	100	40	0	80	20	40	40	20
Ski	0	80	100	20	0	0	40	100	0	60	20	60	0	0
Soccer	100	0	0	0	100	100	0	0	100	0	80	0	0	40
Fencing	0	80	60	20	0	0	80	60	0	100	20	20	20	20
Basketball	80	20	20	0	80	80	20	20	80	20	100	0	0	40
Swimming	0	40	60	60	0	0	40	60	0	20	0	100	40	20
Artistic Gym	0	0	0	80	0	0	40	0	0	20	0	40	100	40
Tennis	40	20	0	40	40	40	20	0	40	20	40	20	40	100

TABLE 2.
Optimal clusters and weights (in parentheses) from the ROOTCLUS model

Cluster 1		Cluster 2		Cluster 3	
Root Cluster 1 (44.75)		Root Cluster 2 (36.54)		Root Cluster 3 (54.67)	
Volleyball Water Polo Rugby Soccer Basketball		Horse-riding Cycling Ski		Athletics Artistic Gymnastic	
Gender-specific Cluster 1		Gender-specific Cluster 2		Gender-specific Cluster 3	
Males (48.10)	Females (35.89)	Males (34.66)	Females (43.75)	Males (39.65)	Females (0)
Tennis	Tennis	Fencing	Fencing Martial Arts Swimming	Martial Arts Swimming	–

Additive constant	
Males	Females
9.57	8.94

The ROOTCLUS model, which will be fully introduced in Sect. 3, has been applied to the Sport Data in Table 1.

The optimal solution (retained as the best one in 200 runs of the algorithm presented in Sect. 4 and attaining a relative[1] loss equal to 0.0824) gives 3 "Root" clusters containing a subset of 10 out of 14 sports that are the *roots* which the *gender-specific* partitions stem from (Table 2). For the sake of completeness, Table 2 also reports the optimal weights and constants from the ROOTCLUS model which will be formalized in Sect. 3 and which are used to estimate the similarities in Table 3 (see Sect. 3 for fully details).

Actually, two additional complementary partitions are found which subsume the similarities expressed by Males and Females, respectively, by differently allocating the remaining four sports to Gender-specific Clusters which join the common Root Clusters (Table 2).

Note that all sports are assigned to some cluster, but while the Root Clusters underly the common perception of the similarities among sports, the *gender-specific* clusters provide a particular connotation of the clusters themselves.

Thus, for example, Cluster 1 identifies the "Ball Sports" including Tennis (Gender-specific Cluster 1) in addition to the "Team Sports" (Root Cluster 1) for both Males and Females. Note that Tennis is not part of the first Root Cluster even if it is a singleton in both gender-specific partitions. In fact, even though both Males and Females perceive Tennis similar to the sports in Root Cluster 1, actually they differ in how they evaluate the strength of such similarity. This is evident from the estimated similarities (Table 3) which confirm that Tennis (the only individual ball sport) belongs to Cluster 1 but it is perceived less similar to the other sports in the cluster (similarity equal to 57.67 and 44.83 for Males and Females, respectively).

[1]The relative loss is defined here as the ratio of the raw loss to the total sum of squares of the data.

The same considerations hold for Fencing: Male students put only Fencing with the sports requiring "Special Equipments" (Root Cluster 2), while Females also add Martial Arts and Swimming. Sports in Root Cluster 2 are very similar according to both Males and Females (similarity equal to 80.77 and 89.23, respectively), but their similarity with Fencing is lower and different for the two genders.

Lastly, Root Cluster 3 is complemented by Males with Martial Arts and Swimming forming the cluster of "Multidisciplinary Sports," whereas Females do not add anything to the Root Cluster: This implies that for Males such two sports are similar to Athletics and Artistic Gym, whereas this is not true for Females.

Moreover, the two sports in Root Cluster 3 are perceived much more similar according to Males than Females (similarity equal to 103.89 and 63.61, respectively).

From Table 3, it is evident that clusters of similar sports have been identified for each gender: A) sports within the same clusters are *always* more similar than sports in different clusters; B) sports belonging to Gender-specific Clusters are *always* more similar to the sports in their corresponding Root Clusters but to a lesser extent than the sports within Root Clusters are to each other.

Finally, as for the assessment of the goodness of the solution, we may consider that 91.76% of the *total* sum of squares is accounted for by the model and can be decomposed into the *within* and *between clusters* sum of squares (88.82% and 2.94%, respectively). Moreover, the *within* sum of squares accounted for can be, in turn, decomposed into the *common within* (67.27%) and *gender within* (21.55%) sum of squares, due to the similarities within Root and Gender-specific Clusters, respectively.

3. The ROOTCLUS Model

Let us assume that a two-mode three-way array $\underline{\mathbf{S}}$ consists of H square matrices $\mathbf{S}_h$ ($h = 1, \ldots, H$) of size N where the entry s_{ilh} represents the pairwise nonnegative similarity between objects i and l ($i, l = 1, \ldots, N$) given by subject h ($h = 1, \ldots, H$).

In order to present the INDCLUS-type model, termed here ROOTCLUS, let us formally recall the INDCLUS model (Carroll and Arabie 1983), which can be written as

$$\mathbf{S}_h = \tilde{\mathbf{P}}\tilde{\mathbf{W}}_h\tilde{\mathbf{P}}' + \tilde{c}_h\mathbf{1}_N\mathbf{1}_N' + \tilde{\mathbf{E}}_h, \qquad (h = 1, \ldots, H), \tag{1}$$

where $\tilde{\mathbf{P}} = [\tilde{p}_{ij}]$ ($\tilde{p}_{ij} = \{0, 1\}$ for $i = 1, \ldots, N$ and $j = 1, \ldots, J$) is the $N \times J$ binary matrix defining the common clustering of the N objects into J possibly overlapping clusters, $\tilde{\mathbf{W}}_h$ is the nonnegative diagonal weight matrix of order J for subject h, $\tilde{c}_h$ is the real-valued additive constant for subject h which can be interpreted as the weight of a "universal cluster" comprising the complete set of the N objects, $\mathbf{1}_N$ denotes the column vector with N ones and $\tilde{\mathbf{E}}_h$ is the square error matrix which is the part of $\mathbf{S}_h$ not accounted for by the model.

Actually, when the subjects present systematic differences, it is reasonable to think that, on the one hand, they can share the same partition only for a *subset* of the whole set of N objects and, on the other hand, they differ in partitioning the remaining ones.

Let us suppose that N_P out of the N objects belong to a common partition (termed here *Root partition*) in J *Root* non-overlapping clusters $\{R_1, \ldots, R_j, \ldots, R_J\}$, while the other $N_M = N - N_P$ objects remain not assigned. In addition, the latter N_M objects are supposed to form H *Individual partitions*, being differently assigned to J *Individual* non-overlapping clusters $\{I_1^h, \ldots, I_j^h, \ldots, I_J^h\}$ ($h = 1, \ldots, H$) by the H subjects. Both $\{R_1, \ldots, R_j, \ldots, R_J\}$ and $\{I_1^h, \ldots, I_j^h, \ldots, I_J^h\}$ ($h = 1, \ldots, H$) define *incomplete* partitions of the N objects because not

TABLE 3.
Sport data: estimated similarities from the ROOTCLUS model

	Volleyball	Horse-riding	Cycling	Athletics	Water Polo	Rugby	Martial Arts	Ski	Soccer	Fencing	Basketball	Swimming	Artistic Gym	Tennis
						Males								
Volleyball	–	9.57	9.57	9.57	102.42	102.42	9.57	9.57	102.42	9.57	102.42	9.57	9.57	57.67
Horse-riding	9.57	–	80.77	9.57	9.57	9.57	9.57	80.77	9.57	44.23	9.57	9.57	9.57	9.57
Cycling	9.57	80.77	–	9.57	9.57	9.57	9.57	80.77	9.57	44.23	9.57	9.57	9.57	9.57
Athletics	9.57	9.57	9.57	–	9.57	9.57	49.22	9.57	9.57	9.57	9.57	49.22	103.89	9.57
Water Polo	102.42	9.57	9.57	9.57	–	102.42	9.57	9.57	102.42	9.57	102.42	9.57	9.57	57.67
Rugby	102.42	9.57	9.57	9.57	102.42	–	9.57	9.57	102.42	9.57	102.42	9.57	9.57	57.67
Martial Arts	9.57	9.57	9.57	49.22	9.57	9.57	–	9.57	9.57	9.57	9.57	49.22	49.22	9.57
Ski	9.57	80.77	80.77	9.57	9.57	9.57	9.57	–	9.57	44.23	9.57	9.57	9.57	9.57
Soccer	102.42	9.57	9.57	9.57	102.42	102.42	9.57	9.57	–	9.57	102.42	9.57	9.57	57.67
Fencing	9.57	44.23	44.23	9.57	9.57	9.57	9.57	44.23	9.57	–	9.57	9.57	9.57	9.57
Basketball	102.42	9.57	9.57	9.57	102.42	102.42	9.57	9.57	102.42	9.57	–	9.57	9.57	57.67
Swimming	9.57	9.57	9.57	49.22	9.57	9.57	49.22	9.57	9.57	9.57	9.57	–	49.22	9.57
Artistic Gym	9.57	9.57	9.57	103.89	9.57	9.57	49.22	9.57	9.57	9.57	9.57	49.22	–	9.57
Tennis	57.67	9.57	9.57	9.57	57.67	57.67	9.57	9.57	57.67	9.57	57.67	9.57	9.57	–

TABLE 3.
continued

	Volleyball	Horse-riding	Cycling	Athletics	Water Polo	Rugby	Martial Arts	Ski	Soccer	Fencing	Basketball	Swimming	Artistic Gym	Tennis
						Females								
Volleyball	–	8.94	8.94	8.94	89.58	89.58	8.94	8.94	89.58	8.94	89.58	8.94	8.94	44.83
Horse-riding	8.94	–	89.23	8.94	8.94	8.94	52.69	89.23	8.94	52.69	8.94	52.69	8.94	8.94
Cycling	8.94	89.23	–	8.94	8.94	8.94	52.69	89.23	8.94	52.69	8.94	52.69	8.94	8.94
Athletics	8.94	8.94	8.94	–	8.94	8.94	8.94	8.94	8.94	8.94	8.94	8.94	63.61	8.94
Water Polo	89.58	8.94	8.94	8.94	–	89.58	8.94	8.94	89.58	8.94	89.58	8.94	8.94	44.83
Rugby	89.58	8.94	8.94	8.94	89.58	–	8.94	8.94	89.58	8.94	89.58	8.94	8.94	44.83
Martial Arts	8.94	52.69	52.69	8.94	8.94	8.94	–	52.69	8.94	52.69	8.94	52.69	8.94	8.94
Ski	8.94	89.23	89.23	8.94	8.94	8.94	52.69	–	8.94	52.69	8.94	52.69	8.94	8.94
Soccer	89.58	8.94	8.94	8.94	89.58	89.58	8.94	8.94	–	8.94	89.58	8.94	8.94	44.83
Fencing	8.94	52.69	52.69	8.94	8.94	8.94	52.69	52.69	8.94	–	8.94	52.69	8.94	8.94
Basketball	89.58	8.94	8.94	8.94	89.58	89.58	8.94	8.94	89.58	8.94	–	8.94	8.94	44.83
Swimming	8.94	52.69	52.69	8.94	8.94	8.94	52.69	52.69	8.94	52.69	8.94	–	8.94	8.94
Artistic Gym	8.94	8.94	8.94	63.61	8.94	8.94	8.94	8.94	8.94	8.94	8.94	8.94	–	8.94
Tennis	44.83	8.94	8.94	8.94	44.83	44.83	8.94	8.94	44.83	8.94	44.83	8.94	8.94	–

all the objects are assigned to either a Root or an Individual cluster. Consequently, for any subject h, the partition of the whole set of the $N = N_p + N_M$ objects can be obtained as the union of the *Root* and the *Individual* partition by assuming $C_j^h = \{R_j \bigcup I_j^h\}$ ($j = 1, \ldots, J$ and $h = 1, \ldots, H$). Note that $\{C_1^h, \ldots, C_j^h, \ldots, C_J^h\}$ ($h = 1, \ldots, H$) defines a *complete* partition of the N objects, i.e., any object is assigned to one and only one cluster by each subject, which is possible because the number of Root clusters and Individual clusters is equal and they are linked one-to-one as detailed in the following.

In order to specify the model, let us set the necessary notation:

- $\mathbf{P} = [p_{ij}]$ ($p_{ij} = \{0, 1\}$ for $i = 1, \ldots, N$ and $j = 1, \ldots, J$) is the $N \times J$ incomplete binary membership matrix defining the *Root partition* of N_P ($\leq N$) objects in J *Root* clusters. Note that $\mathbf{P}$ presents $N_M = N - N_P$ rows of zeros corresponding to the N_M objects not allocated to the Root partition;
- $\mathbf{M}_h = [m_{ijh}]$ ($m_{ijh} = \{0, 1\}$ for $i = 1, \ldots, N$ and $j = 1, \ldots, J$) is the $N \times J$ incomplete binary membership matrix defining the *Individual partition* in J *Individual* clusters of the $N_M = N - N_P$ objects not assigned to $\mathbf{P}$ by subject h ($h = 1, \ldots, H$). Note that $\mathbf{M}_h$ presents N_P rows of zeros and defines a partition complementary to $\mathbf{P}$.

Therefore, the ROOT CLUStering (ROOTCLUS) model can be specified as follows:

$$\mathbf{S}_h = \mathbf{PWP}' + \big(\mathbf{P} + \mathbf{M}_h\big)\mathbf{V}_h\big(\mathbf{P} + \mathbf{M}_h\big)' + c_h\mathbf{1}_N\mathbf{1}_N' + \mathbf{E}_h, \quad (h = 1, \ldots, H), \tag{2}$$

where $\mathbf{W}$ and $\mathbf{V}_h$ are nonnegative diagonal weight matrices of order J, c_h is a real-valued additive constant and $\mathbf{E}_h$ ($h = 1, \ldots, H$) is the error term.

Therefore, on the one hand, we suppose that there exists a subset N_P of the N objects whose pairwise proximities subsume the same partition $\mathbf{P}$ and cluster weights $\mathbf{W}$; on the other hand, subjects are supposed to differ in partitioning the remaining N_M objects providing *Individual partitions* $\mathbf{M}_h$ and weights $\mathbf{V}_h$ ($h = 1, \ldots, H$).

Accordingly, for any subject h ($h = 1, \ldots, H$), the binary membership matrices $\mathbf{P}$ and $\mathbf{M}_h$ identify pairs of *incomplete* and *complementary* partitions so that matrix $(\mathbf{P} + \mathbf{M}_h)$ is a *complete* membership matrix in the sense that no object remains non-assigned and any object is assigned to one and only one cluster of the partition $\{C_1^h, \ldots, C_j^h, \ldots, C_J^h\}$ ($h = 1, \ldots, H$).

The J common clusters $\{R_1, \ldots, R_j, \ldots, R_J\}$ identified by $\mathbf{P}$ can be considered as *Root* clusters in the sense that they are common to all subjects and are the roots at which the *Individual* clusters $\{I_1^h, \ldots, I_j^h, \ldots, I_J^h\}$ identified by $\mathbf{M}_h$ ($h = 1, \ldots, H$) intersect or branch.

For identifiability reasons, weights v_{jh} ($j = 1, \ldots, J$; $h = 1, \ldots, H$) are constrained to be null when the corresponding *Individual* cluster is empty ($I_j^h = \emptyset$).

Remark 1. Up to the constants c_h, weight w_j ($j = 1, \ldots, J$) represents the similarity between objects in the *Root* cluster R_j for all subjects and measures the strength of the "baseline" similarity between the N_P objects belonging to R_j, while weight v_{jh} ($j = 1, \ldots, J$; $h = 1, \ldots, H$) measures the strength of the similarity between objects in the *Individual* cluster I_j^h. Thus, for any subject h the estimated similarities for objects in cluster $C_j^h = \{R_j \bigcup I_j^h\}$ are finally larger between objects within R_j (equal to $w_j + v_{jh}$), while they are weaker (and equal to v_{jh}) between pairs of objects a) both in I_j^h or b) one in R_j and one in I_j^h. The idea behind is that if subject h has object i in his *Individual* cluster I_j^h, whereas this is not true for object i', this implies that i is more similar than i' to objects in *Root* cluster R_j.

3.1. Additional Requirements

The key assumption here is that the subject heterogeneity of the three-way similarities cannot be explained only by a single partition of the objects common to all subjects, but subsumes also Individual partitions which link to the common one. In such a respect, we may always regard that for each subject h, any cluster C_j^h of his complete partition is composed by the subset R_j, which is common to all subjects, and by the complementary subset I_j^h, which is subject specific.

For any $C_j^h = \{R_j \bigcup I_j^h\}$ ($j = 1, \ldots, J$ and $h = 1, \ldots, H$), different cases can be considered:

a) $R_j \neq \emptyset$ and $I_j^h = \emptyset \Rightarrow C_j^h \neq \emptyset$: subject h assigns no objects to I_j^h, which means that no objects are regarded similar to those in R_j;
b) $R_j \neq \emptyset$ and $I_j^h \neq \emptyset \Rightarrow C_j^h \neq \emptyset$: subject h assigns some objects to I_j^h because they are similar to objects in R_j but less similar than the objects within R_j are to each other;
c) $R_j = \emptyset$ and $I_j^h = \emptyset \Rightarrow C_j^h = \emptyset$: subject h does not assign any object to both Root and Individual clusters j;
d) $R_j = \emptyset$ and $I_j^h \neq \emptyset \Rightarrow C_j^h \neq \emptyset$: subject h assigns some objects to I_j^h even if R_j is empty.

Note that, regardless of the assumption of non-overlapping clusters, when case a) is true for all subjects and clusters, it corresponds to fit the INDCLUS model (1) which searches for the same clusters R_j ($j = 1, \ldots, J$) across subjects. Again, when case d) is true for all subjects and clusters, it corresponds to performing ADCLUS (Shepard and Arabie 1979) to each $\mathbf{S}_h$ ($h = 1, \ldots, H$) separately. On the one hand, the latter allows for the maximum flexibility because there are no objects in the common clusters and all objects belong to the individual ones; on the other hand, the resulting subject-specific clusters are obtained independently and are not linked across subjects.

The ROOTCLUS model covers here cases a), b), c) while case d) is assumed not admissible because the aim is to identify clusters R_j ($j = 1, \ldots, J$) which are linked one-to-one to the corresponding Individual clusters I_j^h ($j = 1, \ldots, J$; $h = 1, \ldots, H$). Thus, the common meaning of all C_j^h ($j = 1, \ldots, J$; $h = 1, \ldots, H$) is given by their intersection set R_j which subsumes the underlying core of similarities shared by all subjects.

Thus, some additional constraints are required to specify the model. Hence, case d) is avoided by controlling that for any pair of subjects h and h' ($h \neq h'$; $h, h' = 1, \ldots, H$) for any cluster j ($j = 1, \ldots, J$): $C_j^h \bigcap C_j^{h'} = R_j = \emptyset \Rightarrow I_j^h = I_j^{h'} = \emptyset$.

This is equivalent to set the following additional constraint for the membership matrices $\mathbf{P}$ and $\mathbf{M}_h$,

$$\text{if} \quad \sum_{i=1}^{N} p_{ij} = 0 \quad \text{then} \quad \sum_{i=1}^{N} m_{ijh} = 0, \quad (j = 1, \ldots, J\,;\; h = 1, \ldots, H) \tag{3}$$

where no Individual non-empty clusters are assumed admissible whenever the corresponding common cluster is empty.

Furthermore, the more objects belong to the Root cluster R_j, the more parsimonious the model is; conversely, when the common subset R_j contains just a few objects, the Individual clusters I_j^h are more flexible in accounting for the observed similarities. Thus, as a second requirement a trade-off is necessary between the need for parsimony (as many objects allocated to common partition as possible) and the goodness of fit (a large number of objects N_M differently allocated by each subject). Without any other constraint, the best model may lead to find the more flexible solution by allocating the minimum number of objects to the Root clusters.

That is why, we need an additional requirement to control the non-sparsity of matrix $\mathbf{P}$ and, equivalently, the sparsity of matrices $\mathbf{M}_h$ ($h = 1, \ldots, H$) in order to force toward solutions where the number of objects assigned to the Root partition is as large as possible:

$$G = \frac{N - N_P}{N_P^2} = \frac{1}{N_P}\left(\frac{N_M}{N_P}\right) \leq s \tag{4}$$

where s is a specified parameter.

Since G in (4) is a function of the ratio between the number of objects non-allocated N_M and allocated N_P to the Root clusters, respectively, it can be used to penalize the model. Note that the function G is 0 when all objects belong to the Root partition ($N_P = N$), while G increases as the number of objects in any Individual partition increases and, to accelerate such an increase, N_P^2 is used at the denominator instead of N_P.

Remark 2. Note that the ROOTCLUS model could be reformulated in terms of overlapping clusters (with appropriate modifications of the algorithm presented in the following Sect. 4), but with an additional burden in terms of model complexity. Actually, this is not a limitation in most situations because the ROOTCLUS model is rather flexible as it allows to assign any object to different Individual clusters. Since Root and Individual clusters are linked one-to-one, clusters C_j^h ($h = 1, \ldots, H$) synthesize the subject heterogeneity by combining the common Root cluster R_j with the Individual clusters I_j^h ($h = 1, \ldots, H$). Thus, objects in Root clusters are evaluated similar by *all* subjects and assigned to only one cluster, while the remaining objects may belong to more than one cluster C_j^h ($h = 1, \ldots, H$) according to how subjects differently evaluate or perceive the pairwise proximities (see illustrative example in Sect. 3.2 and applications in Sect. 6.2).

3.2. Illustrative Example: Artificial Data

In order to better illustrate the meaning of any part of the model, let us consider the example in Table 4, where artificial three-way similarity data are displayed for $N = 10$ objects, $H = 3$ subjects and $J = 3$ clusters, together with the membership matrices $\mathbf{P}$ and $\mathbf{M}_h$ ($h = 1, \ldots, H$).

The pattern underlying the data is evident. The heat map of each similarity matrix shows 3 "green blocks" on the main diagonal having common (dark green) "roots" across matrices: namely, $R_1 = (A, B, C)$, $R_2 = (E, F)$, $R_3 = (L, M)$, corresponding to the incomplete membership matrix $\mathbf{P}$.

Actually, the remaining three objects D, G, K differently frame the blocks across the subjects. For example, object D frames the first block for both subject S_1 and S_3, while for subject S_2 object D joins the second block together with object G (see also the membership matrices $\mathbf{M}_1, \mathbf{M}_2, \mathbf{M}_3$). The different shades of green show how strong the similarities are within *Root* and *Individual* clusters. Thus, up to the individual constants c_h (which play the same role as in INDCLUS), the weight w_j ($j = 1, 2, 3$) provides the baseline similarity between objects within each block (*Root* cluster R_j) across subjects, which is possibly augmented by v_{jh} depending on whether subject S_h adds other objects to cluster j. The individual weight v_{jh} represents the average similarity between objects due to the *Individual* cluster j. For instance, objects G and K belong to cluster 2 in S_1 and cluster 3 in S_3 with similarities given by $s_{GK1} = v_{21} + c_1 = 3 + 1 = 4$ and $s_{GK3} = v_{33} + c_3 = 1 + 1 = 2$, respectively; conversely, they belong to different clusters in S_2 with smaller similarity $s_{GK2} = c_2 = 1$.

For the sake of completeness, while the ROOTCLUS model fits perfectly the data (relative loss equal to zero), the INDCLUS model fitted to such data attains a relative loss equal to 0.0722 corresponding to the optimal matrices $\tilde{\mathbf{P}}$ and $\tilde{\mathbf{W}}$ given in Table 5. It is evident that although the unique classification identified by matrix $\tilde{\mathbf{P}}$ is quite flexible because it allows for overlapping

TABLE 4.
Artificial three-way similarity data (Color table online)

S_1	A	B	C	D	E	F	G	K	L	M
A	8	8	8	5	1	1	1	1	1	1
B	8	8	8	5	1	1	1	1	1	1
C	8	8	8	5	1	1	1	1	1	1
D	5	5	5	5	1	1	1	1	1	1
E	1	1	1	1	6	6	4	4	1	1
F	1	1	1	1	6	6	4	4	1	1
G	1	1	1	1	4	4	4	4	1	1
K	1	1	1	1	4	4	4	4	1	1
L	1	1	1	1	1	1	1	1	8	8
M	1	1	1	1	1	1	1	1	8	8

M_1				
A	0	0	0	v_{11}=4
B	0	0	0	v_{21}=3
C	0	0	0	v_{31}=0
D	1	0	0	
E	0	0	0	c_1=1
F	0	0	0	
G	0	1	0	
K	0	1	0	
L	0	0	0	
M	0	0	0	

P			
A	1	0	0
B	1	0	0
C	1	0	0
D	0	0	0
E	0	1	0
F	0	1	0
G	0	0	0
K	0	0	0
L	0	0	1
M	0	0	1

w_1=3
w_2=2
w_3=7

S_2	A	B	C	D	E	F	G	K	L	M
A	4	4	4	1	1	1	1	1	1	1
B	4	4	4	1	1	1	1	1	1	1
C	4	4	4	1	1	1	1	1	1	1
D	1	1	1	7	7	7	1	1	1	1
E	1	1	1	7	9	9	7	1	1	1
F	1	1	1	7	9	9	7	1	1	1
G	1	1	1	7	7	7	7	1	1	1
K	1	1	1	1	1	1	1	3	3	3
L	1	1	1	1	1	1	1	3	10	10
M	1	1	1	1	1	1	1	3	10	10

M_2				
A	0	0	0	v_{12}=0
B	0	0	0	v_{22}=6
C	0	0	0	v_{32}=2
D	0	1	0	
E	0	0	0	c_2=1
F	0	0	0	
G	0	1	0	
K	0	0	1	
L	0	0	0	
M	0	0	0	

$$\frac{1}{H}\sum_{h=1}^{H}(\mathbf{P}+\mathbf{M}_h)$$

A	1	0	0
B	1	0	0
C	1	0	0
D	**2/3**	**1/3**	0
E	0	1	0
F	0	1	0
G	0	**2/3**	**1/3**
K	0	**1/3**	**2/3**
L	0	0	1
M	0	0	1

S_3	A	B	C	D	E	F	G	K	L	M
A	7	7	7	4	1	1	1	1	1	1
B	7	7	7	4	1	1	1	1	1	1
C	7	7	7	4	1	1	1	1	1	1
D	4	4	4	4	1	1	1	1	1	1
E	1	1	1	1	3	3	1	1	1	1
F	1	1	1	1	3	3	1	1	1	1
G	1	1	1	1	1	1	2	2	2	2
K	1	1	1	1	1	1	2	2	2	2
L	1	1	1	1	1	1	2	2	9	9
M	1	1	1	1	1	1	2	2	9	9

M_3				
A	0	0	0	v_{13}=3
B	0	0	0	v_{23}=0
C	0	0	0	v_{33}=1
D	1	0	0	
E	0	0	0	c_3=1
F	0	0	0	
G	0	0	1	
K	0	0	1	
L	0	0	0	
M	0	0	0	

clusters, it does not manage to capture all the subject heterogeneity. Object D is correctly assigned to both clusters 1 and 2; object K remains not assigned to any cluster (which means that K is not similar to any other object), while G belongs to the second cluster only.

Conversely, while ROOTCLUS does not allow for overlap of the Root clusters (A, B, C, E, F, L, M belong to only one cluster), it actually captures the subject heterogeneity through matrices $\mathbf{M}_h$, so that matrix $\frac{1}{H}\sum_{h=1}^{H}(\mathbf{P}+\mathbf{M}_h)$ (see Table 4) synthesizes how each object either belongs to only one Root cluster or to several Individual clusters with different proportions due to subjects. For example, object D belongs to the *first* and *second* clusters with proportions 2/3 and 1/3, respectively, which correctly reproduces the data in Table 4. Note that this is meaningful because of the one-to-one link between Root and Individual clusters.

TABLE 5.
Artificial data: results from INDCLUS

	$\tilde{\mathbf{P}}$				$\tilde{\mathbf{W}}$		
A	1	0	0	S_1	5.20	1.74	6.63
B	1	0	0	S_2	1.49	6.22	8.82
C	1	0	0	S_3	4.47	0.27	7.80
D	1	1	0	Constant	1.18	1.37	1.20
E	0	1	0				
F	0	1	0				
G	0	1	0				
K	0	0	0				
L	0	0	1				
M	0	0	1				

3.3. The Least-Squares Problem

In order to achieve our goal of partitioning the objects of the three-way similarity data, we specify the least-squares estimation of model (2) as the solution of the following constrained problem

$$\begin{aligned} &\min\ F(\mathbf{P}, \mathbf{W}, \mathbf{M}_h, \mathbf{V}_h, c_h) \\ &= \frac{\sum_{h=1}^{H} \left\| \mathbf{S}_h - \mathbf{PWP}' - \left(\mathbf{P}+\mathbf{M}_h\right)\mathbf{V}_h\left(\mathbf{P}+\mathbf{M}_h\right)' - c_h \mathbf{1}_N \mathbf{1}_N' \right\|^2}{\sum_{h=1}^{H} \|\mathbf{S}_h\|^2} \end{aligned} \tag{5}$$

subject to

$$p_{ij} = \{0, 1\} \quad (i = 1, \ldots, N;\ j = 1, \ldots, J),\ \sum_{j=1}^{J} p_{ij} \leq 1 \quad (i = 1, \ldots, N), \tag{6}$$

$$m_{ijh} = \{0, 1\} \quad (i = 1, \ldots, N;\ j = 1, \ldots, J;\ h = 1, \ldots, H),$$

$$\sum_{j=1}^{J} m_{ijh} \leq 1 \quad (i = 1, \ldots, N;\ h = 1, \ldots, H), \tag{7}$$

$$\sum_{j=1}^{J} (p_{ij} + m_{ijh}) = 1 \quad (i = 1, \ldots, N;\ h = 1, \ldots, H), \tag{8}$$

$$\forall\, j\ :\ \sum_{i=1}^{N} p_{ij} = 0\ \Rightarrow\ \sum_{i=1}^{N} m_{ijh} = 0 \quad (j = 1, \ldots, J;\ h = 1, \ldots, H), \tag{9}$$

$$w_j \geq 0 \quad (j = 1, \ldots, J), \tag{10}$$

$$v_{jh} \geq 0 \quad (j = 1, \ldots, J;\ h = 1, \ldots, H), \tag{11}$$

$$\forall\, j\ :\ \sum_{i=1}^{N} m_{ijh} = 0\ \Rightarrow\ v_{jh} = 0 \quad (h = 1, \ldots, H), \tag{12}$$

$$G = \frac{N - \sum_{i=1}^{N} \sum_{j=1}^{J} p_{ij}}{(\sum_{i=1}^{N} \sum_{j=1}^{J} p_{ij})^2} \ \leq\ s. \tag{13}$$

Specifically, the set of constraints (6)–(8) specify that in the partitions determined by $\mathbf{P}$ and $\mathbf{M}_h$ ($h = 1, \ldots, H$) each object is assigned to only one cluster in either $\mathbf{P}$ or $\mathbf{M}_h$, while constraints (9) and (13) are equivalent to (3) and (4), respectively.

Since there is a one-to-one correspondence between s in (13) and the nonnegative parameter λ in the following function:

$$\min f(\mathbf{P}, \mathbf{W}, \mathbf{M}_h, \mathbf{V}_h, c_h) = \min \{F(\mathbf{P}, \mathbf{W}, \mathbf{M}_h, \mathbf{V}_h, c_h) + \lambda G\}, \tag{14}$$

the constrained problem (5) can be equivalently formulated as the minimization of (14) subject to the set of constraints (6)–(13). Note that, the higher value of λ, the stronger the penalty G.

An appropriate efficient Alternating Least-Squares (ALS)-type algorithm is presented in Sect. 4.

4. The Algorithm

The constrained problem (14) can be solved by using an Alternating Least-Squares (ALS)-type algorithm, which is defined as a block-relaxation algorithm (de Leeuw 1994) applied to a least-squares loss function. Thus, the complex optimization problem (14) is solved by alternatingly updating each block of parameters while maintaining all the others fixed as detailed as follows.

Similar to the schemes employed in SINDCLUS (Chaturvedi and Carroll 1994) and SYMPRES (Kiers 1997) for fitting the INDCLUS model, given J and λ, an ALS-type algorithm for fitting the ROOTCLUS model is presented:

Step 0 *Initialization*.
Step 1 *Updating matrices* $\mathbf{P}$ and $\mathbf{M}_h$ ($h = 1, \ldots, H$): both matrices $\mathbf{P}$ and $\mathbf{M}_h$ ($h = 1, \ldots, H$) are updated together in a row-wise fashion by solving assignment problems.
Step 2 *Updating weight matrix* $\mathbf{W}$: weight matrix $\mathbf{W}$ is estimated as the solution of a constrained regression problem.
Step 3 *Updating weight matrix* $\mathbf{V}_h$ ($h = 1, \ldots, H$): weight matrices $\mathbf{V}_h$ ($h = 1, \ldots, H$) are estimated by solving H independent constrained regression models.
Step 4 *Updating constant* c_h ($h = 1, \ldots, H$): individual constants c_h ($h = 1, \ldots, H$) are estimated by successive residualizations of the data matrix.
Step 5 *Stopping rule*.

The four main steps 1 to 4 are alternated and iterated until convergence. The loss function (14) cannot increase at each step, and the algorithm stops when the loss decreases less than a fixed arbitrary positive and small threshold.

In order to increase the chance of finding the global minimum, the best solution over different random starting parameters is retained.

Without loss of generality and just for simplicity of the notation, in the following, we assume here that the diagonal entries of the H similarity matrices $\mathbf{S}_h$ are fitted; the case where the diagonal entries are not of interest can be derived straightforwardly (see "Appendix").

A detailed description of the steps of the algorithm, implemented in MATLAB R2017b,[2] follows.

Step 0 *Initialization.*

Initial estimates of the parameters $\hat{\mathbf{P}}$, $\hat{\mathbf{W}}$, $\hat{\mathbf{M}}_h$, $\hat{\mathbf{V}}_h$ and $\hat{c}_h$ ($h = 1, \ldots, H$) are chosen randomly or in a rational way, but they are required to satisfy the set of constraints (6)–(13).

To improve the chance of finding a global minimum, in the simulation study and applications, a rational starting $\hat{\mathbf{P}}$ has been chosen by *a*) taking the absolute values of the first J eigenvectors of the mean similarity matrix $\frac{1}{H}\sum_{h=1}^{H}\mathbf{S}_h$ and *b*) setting all row-wise highest elements to 1 and

[2]The MATLAB code is available online on SpringerLink with the article.

all other elements to 0. Afterward, for $J < N-1$ a total of J rows have been randomly set to zero, while to get feasible starting solutions for either $J = N-1$ or $J = N$ a total of $N-2$ rows have been randomly set to zero.

Alternative rational starts could be differently obtained starting from the INDCLUS solution or from a separate ADCLUS analysis per subject, and deriving appropriate feasible starting solutions.

Step 1 *Updating matrices* $\mathbf{P}$ and $\mathbf{M}_h$ $(h = 1, \ldots, H)$.

Given the current estimates of $\hat{\mathbf{W}}$, $\hat{\mathbf{V}}_h$ and $\hat{c}_h$ $(h = 1, \ldots, H)$, problem (14) is solved over $\mathbf{P}$ and $\mathbf{M}_h$ $(h = 1, \ldots, H)$ for each row i $(i = 1, \ldots, N)$ by solving assignment problems which minimize (14).

In the following, let $\mathbf{0}$ be the J-column vector of zeros, $\mathbf{u}_j$ be the J-dimensional vector with all entries equal to 0 except for the j-th which is 1, n_j be the number of objects in Root cluster j, $\mathbf{p}_i$ be the i-th row of $\mathbf{P}$, $\mathbf{m}_{ih}$ be the i-th row of $\mathbf{M}_h$, $f(a)$ be the loss function (14) computed using the entity within parentheses.

The updating of the i-th rows of $\mathbf{P}$ and $\mathbf{M}_h$ $(h = 1, \ldots, H)$, while keeping all the other rows fixed, is described in pseudo-code as follows.

A. Main loop. For $i = 1, \ldots, N$

B. For $j = 1, \ldots, J+1$

B1. If $j < J+1$

b_{11}) Set $\mathbf{p}_i = \mathbf{u}_j$

b_{12}) Set $\mathbf{m}_{ih} = \mathbf{0}$ $(h = 1, \ldots, H)$

b_{13}) Compute $f^j = f\left(\mathbf{p}_i, \mathbf{m}_{i1}, \ldots, \mathbf{m}_{ih}, \ldots, \mathbf{m}_{iH}\right)$

B2. Else if $j = J+1$

b_{21}) Set $\mathbf{p}_i = \mathbf{0}$

b_{22}) Compute $J^* = \text{card}\left(\{n_1, \ldots, n_j, \ldots, n_J \ : \ n_j > 0\}\right)$

C. Inner loop. For $h = 1, \ldots, H$

D. For $j = 1, \ldots, J$

D1. If $n_j > 0$

d_{11}) Set $\mathbf{m}_{ih} = \mathbf{u}_j$

d_{12}) Compute $f^{hj} = f\left(\mathbf{p}_i, \mathbf{m}_{ih}\right)$

END D

C1. Select $l^h = \underset{1 \le j \le J^*}{\text{argmin}} \left(f^{hj}\right)$

C2. Set $\mathbf{m}_{ih} = \mathbf{u}_{l^h}$

END C

b_{23}) Compute $f^{J+1} = f\left(\mathbf{p}_i, \mathbf{m}_{i1}, \ldots, \mathbf{m}_{ih}, \ldots, \mathbf{m}_{iH}\right)$

END B

E. Select $\upsilon = \underset{1 \le j \le (J+1)}{\text{argmin}} \left\{f^j\right\}$

F1. If $1 \le \upsilon \le J$

f_{11}) Set $\hat{\mathbf{p}}_i = \mathbf{u}_\upsilon$, $\hat{\mathbf{m}}_{ih} = \mathbf{0}$ $(h = 1, \ldots, H)$

F2. Else if $\upsilon = J+1$

f_{21}) Set $\hat{\mathbf{p}}_i = \mathbf{0}$, $\hat{\mathbf{m}}_{ih} = \mathbf{u}_{l^h}$ $(h = 1, \ldots, H)$

END A

Briefly, any object i may be either assigned to $\mathbf{P}$, i.e., to one Root cluster $\{R_1, \ldots, R_j, \ldots, R_J\}$, or remained non-assigned. In the latter case, the updating of the i-th row of $\mathbf{M}_h$ is achieved by solving J^* ($\leq J$) standard assignment problems (Inner loop), being J^* the number of non-empty columns in the current $\mathbf{P}$. This guarantees that constraint (9) is fulfilled.

Before describing steps 2 to 4, it is worth noting that model (2) can be rewritten as

$$\mathbf{s}_h = \mathbf{T}\mathbf{w} + \big(\mathbf{T} + \mathbf{Q}_h\big)\mathbf{v}_h + c_h\mathbf{1}_{N^2} + \mathbf{e}_h, \qquad (h = 1, \ldots, H), \tag{15}$$

where

- $\mathbf{s}_h$ ($h = 1, \ldots, H$) is the column vector of size N^2 of the vectorized matrix $\mathbf{S}_h$, i.e., $\mathbf{s}_h = vec(\mathbf{S}_h) = [s_{11h}, \ldots, s_{1Nh}, \ldots, s_{N1h}, \ldots, s_{NNh}]'$;
- $\mathbf{T}$ is the $N^2 \times J$ matrix formed by the Khatri–Rao product[3] (McDonald 1980; Rao and Mitra 1971) of $\mathbf{P}$ with itself, i.e., $\mathbf{T} = \mathbf{P}| \otimes |\mathbf{P}$;
- $\mathbf{Q}_h$ ($h = 1, \ldots, H$) is the $N^2 \times J$ matrix obtained as $\mathbf{Q}_h = \big(\mathbf{M}_h \,|\otimes|\, (\mathbf{P} + \mathbf{M}_h)\big) + \big(\mathbf{P} \,|\otimes|\, \mathbf{M}_h\big)$;
- $\mathbf{w}$ is the column vector with the J diagonal entries of $\mathbf{W}$;
- $\mathbf{v}_h$ ($h = 1, \ldots, H$) is the column vector with the J diagonal entries of $\mathbf{V}_h$;
- $\mathbf{e}_h$ ($h = 1, \ldots, H$) is the column vector of size N^2 of the errors.

Therefore, by taking into account (15), function (5) becomes

$$\min \; F(\mathbf{P}, \mathbf{W}, \mathbf{M}_h, \mathbf{V}_h, c_h) = \frac{\sum_{h=1}^{H} \left\| \mathbf{s}_h - \mathbf{T}\mathbf{w} - \big(\mathbf{T} + \mathbf{Q}_h\big)\mathbf{v}_h - c_h\mathbf{1}_{N^2} \right\|^2}{\sum_{h=1}^{H} \|\mathbf{s}_h\|^2}. \tag{16}$$

Thus, given the current estimates of $\hat{\mathbf{P}}$ and $\hat{\mathbf{M}}_h$ ($h = 1, \ldots, H$), the minimization of (14) over $\mathbf{W}$, $\mathbf{V}_h$ and c_h ($h = 1, \ldots, H$) is achieved by minimizing (16) subject to

$$\mathbf{w} \geq \mathbf{0} \tag{17}$$

$$\mathbf{v}_h \geq \mathbf{0}; \quad \forall j : \sum_{i=1}^{N} m_{ijh} = 0 \;\Rightarrow\; v_{jh} = 0, \qquad (h = 1, \ldots, H). \tag{18}$$

Step 2 *Updating matrix* $\mathbf{W}$.
Given the current estimates of $\hat{\mathbf{P}}, \hat{\mathbf{M}}_h, \hat{\mathbf{V}}_h$ and $\hat{c}_h$ ($h = 1, \ldots, H$), the estimation of $\mathbf{W}$ is obtained as the solution of the regression problem (16) over $\mathbf{w}$ subject to constraint (17) by using nonnegative least-squares (Lawson and Hanson 1974).
Step 3 *Updating matrix* $\mathbf{V}_h$ ($h = 1, \ldots, H$).
Given the current estimates of $\hat{\mathbf{P}}$, $\hat{\mathbf{W}}$, $\hat{\mathbf{M}}_h$ and $\hat{c}_h$ ($h = 1, \ldots, H$), it is readily seen that minimizing (16) over $\mathbf{v}_h$ ($h = 1, \ldots, H$) is equivalent to minimize each term $F_h(\mathbf{v}_h) = \left\| \mathbf{s}_h - \hat{\mathbf{T}}\hat{\mathbf{w}} - \big(\hat{\mathbf{T}} + \hat{\mathbf{Q}}_h\big)\mathbf{v}_h - \hat{c}_h\mathbf{1}_{N^2} \right\|^2$, subject to constraint (18), separately. The estimation is obtained as the solution of a constrained regression problem by using nonnegative least-squares (Lawson and Hanson 1974). This procedure is iterated H times, and constraint (18) is imposed in the case of an empty Individual cluster.

[3] Given two matrices $\mathbf{A}$ and $\mathbf{B}$ with the same number J of columns, the Khatri–Rao product of $\mathbf{A}$ and $\mathbf{B}$ is the column-wise Kronecker product, i.e., $\mathbf{A}| \otimes |\mathbf{B} = (\mathbf{a}_1 \otimes \mathbf{b}_1, \ldots, \mathbf{a}_j \otimes \mathbf{b}_j, \ldots, \mathbf{a}_J \otimes \mathbf{b}_J)$ where $\mathbf{a}_j$ and $\mathbf{b}_j$ are the j-th ($j = 1, \ldots, J$) column of $\mathbf{A}$ and $\mathbf{B}$, respectively, and $\otimes$ denotes the Kronecker product.

Step 4 *Updating constant* c_h $(h = 1, \ldots, H)$.
Given the current estimates of $\hat{\mathbf{P}}$, $\hat{\mathbf{W}}$, $\hat{\mathbf{M}}_h$, $\hat{\mathbf{V}}_h$ $(h = 1, \ldots, H)$, the estimate of c_h is given by

$$c_h = \frac{\mathbf{1}'_{N^2}\left(\mathbf{s}_h - \hat{\mathbf{T}}\hat{\mathbf{w}} - \left(\hat{\mathbf{T}} + \hat{\mathbf{Q}}_h\right)\hat{\mathbf{v}}_h\right)}{N^2} \qquad (h = 1, \ldots, H).$$

Step 5 *Stopping rule*.
The loss function value is computed for the current values of $\hat{\mathbf{P}}$, $\hat{\mathbf{W}}$, $\hat{\mathbf{M}}_h$, $\hat{\mathbf{V}}_h$ and $\hat{c}_h$ $(h = 1, \ldots, H)$ and since $f\left(\hat{\mathbf{P}}, \hat{\mathbf{W}}, \hat{\mathbf{M}}_h, \hat{\mathbf{V}}_h, \hat{c}_h\right)$ is bounded from below, it converges to a point which is expected to be at least a local minimum. When the function f has not decreased considerably with respect to a convergence tolerance value, the process is assumed to be converged. Otherwise, steps 1 to 4 are repeated in turn.

4.1. Model Assessment and Selection

4.1.1. Choice of the Number of Clusters It is important to notice that the algorithm to fit the ROOTCLUS model does not need to be run by varying the number of clusters, but setting J equal to the maximum number expected or preferred is enough, so that the choice of the optimal number of clusters is not a critical issue here. In fact, the presence of empty clusters (corresponding to columns of zeros both in $\mathbf{P}$ and $\mathbf{M}_h$, $h = 1, \ldots, H$) does not affect the goodness of the solution and the number of non-empty clusters J_P is assumed to be the optimal one.

A general criterion to set the maximum number of clusters cannot be given, but it depends on the specific application. Generally, a number of clusters J large enough can be set to get $J_p < J$ non-empty clusters or, when J_P is too large for practical use, the scree plot of the loss values of the model with different numbers of clusters can be analyzed.

4.1.2. Model Assessment In order to assess different aspects of the fit of the ROOTCLUS model, it can be useful to provide a decomposition of the total variability where the different parts due to Root and Individual partitions are evaluated and can be possibly used to better analyze and choice the optimal solution. Without loss of generality, we consider here that once the ROOTCLUS model has been fitted, we may decompose the total variability starting from (15),

$$\begin{aligned} TSS &= RSS + ESS \\ \sum_{h=1}^{H} \|\mathbf{s}_h\|^2 &= \sum_{h=1}^{H} \left\| \hat{\mathbf{T}}\hat{\mathbf{w}} + \left(\hat{\mathbf{T}} + \hat{\mathbf{Q}}_h\right)\hat{\mathbf{v}}_h + \hat{c}_h \mathbf{1}_{N^2} \right\|^2 \\ &\quad + \sum_{h=1}^{H} \left\| \hat{\mathbf{s}}_h - \hat{\mathbf{T}}\hat{\mathbf{w}} - \left(\hat{\mathbf{T}} + \hat{\mathbf{Q}}_h\right)\hat{\mathbf{v}}_h - \hat{c}_h \mathbf{1}_{N^2} \right\|^2 \end{aligned} \qquad (19)$$

where TSS is the Total Sum of Squares, while RSS and ESS denote the ROOTCLUS and the Error Sum of Squares, respectively.

Note that, once the ROOTCLUS model has been estimated, given λ, RSS is actually the part of the observed similarities accounted for by the model and, specifically, the variability due to clusters $(\hat{\mathbf{P}} + \hat{\mathbf{M}}_h)$ $(h = 1, \ldots, H)$. Additionally, since $(\hat{\mathbf{P}} + \hat{\mathbf{M}}_h)$ is a full standard membership matrix, for any h the variability accounted for by such complete partition can be further decomposed in its Within and Between parts, respectively

$$RSS = WSS + BSS$$
$$= \sum_{h=1}^{H} \left\| \hat{\mathbf{T}}\hat{\mathbf{w}} + (\hat{\mathbf{T}} + \hat{\mathbf{Q}}_h)(\hat{\mathbf{v}}_h + \hat{c}_h \mathbf{1}_J) \right\|^2 + \sum_{h=1}^{H} \left\| \left(\mathbf{1}_{N^2} - (\hat{\mathbf{T}} + \hat{\mathbf{Q}}_h)\mathbf{1}_J\right)\hat{c}_h \right\|^2. \quad (20)$$

Moreover, the WSS can be further decomposed by taking into account the part due to the *Root* (common) *partition* $\hat{\mathbf{P}}$ and the *Individual partitions* $\hat{\mathbf{M}}_h$ $(h = 1, \ldots, H)$

$$WSS = CWSS + IWSS$$
$$= \sum_{h=1}^{H} \left\| \hat{\mathbf{T}}(\hat{\mathbf{w}} + \hat{\mathbf{v}}_h + \hat{c}_h \mathbf{1}_J) \right\|^2 + \sum_{h=1}^{H} \left\| \hat{\mathbf{Q}}_h(\hat{\mathbf{v}}_h + \hat{c}_h \mathbf{1}_J) \right\|^2. \quad (21)$$

Actually, $CWSS$ is the part of the within variability due to the objects belonging to the partition which is common to all subjects, while $IWSS$ is the part accounted for by the Individual partitions.

Thus, plugging (21) into (20) and then into (19), it holds

$$TSS = (CWSS + IWSS + BSS) + ESS.$$

For the sake of interpretability, all the Sum of Squares can be related to the TSS of the similarity data so that ESS/TSS is actually the relative loss (5) and, similarly, $CWSS/TSS$ and $IWSS/TSS$ are relative measures to assess how well the Root and Individual partitions account for the observed similarities.

4.1.3. Model Selection In order to control the non-sparsity of the (incomplete) membership matrix $\mathbf{P}$ (Sect. 3.1), the tuning parameter s has been introduced in (4) which controls the number of objects N_P belonging to the Root partition. Note that because of (9), the solution where $N_P = 0$ (empty matrix $\mathbf{P}$) produces also empty Individual partitions (empty matrices $\mathbf{M}_h$) and the ROOTCLUS model reduces to $\mathbf{S}_h = c_h \mathbf{1}_N \mathbf{1}'_N + \mathbf{E}_h$ which is the trivial solution where each subject is modeled independently without any clustering structure. This trivial solution can be actually avoided by setting a finite value for s in (4) or equivalently $\lambda > 0$ in (14). In fact, since G gets larger as the number of objects N_P becomes smaller, the second term in (14) forces toward solutions with at least $N_P = 1$ even for small values of λ. Thus, matrix $\mathbf{P}$ (matrices $\mathbf{M}_h$) will be non-sparse (sparse) for an appropriate choice of the tuning parameter λ.

When applied to a dataset as a descriptive method, in order to obtain an intuitive understanding of the data, one might simply fix the tuning parameter based on some appropriate criterion: For instance, one could select large values of λ (small values of s), in order to obtain matrices $\mathbf{P}$ having a desirable level of non-sparsity, or determine the tuning parameter λ such that N_P equals a certain value by trial and error.

Conversely, in a confirmatory perspective, the question is which solution should be preferred across different values of λ in order to take a trade-off between a large number of objects in the common partition and an acceptable lack of fit due to such a constraint. In such a respect, the analysis of the scree plot of the loss values across the λ values may help to identify the minimum λ which does not display a sizeable increase in the loss (usually corresponding to an elbow). In addition, to ease the model selection in terms of choice of λ we propose to make use of an index which relies on the decomposition (18) and measures the quality of the solution taking into account the model complexity. Similar to the pseudo-F index, Calinski and Harabasz (1974)

proposed and used in different contexts (Rocci and Vichi 2008; Schepers et al. 2008) just to select the correct model, we consider the ratio of the variability accounted for by the model (RSS) to the residual variability (ESS), both corrected for the degrees of freedom:

$$pF = \frac{RSS/dR}{ESS/dE} \tag{22}$$

where the corresponding degrees of freedom are: 1) $dR = N_P + J_P + J_M + H + H(N - N_P)$ with J_M equal to the total number of non-empty Individual clusters and 2) $dE = (HN(N+1)/2) - dR$ when the diagonal entries are fitted, otherwise $dE = (HN(N-1)/2) - dR$.

The performance of pF in the simulation study is reported in Tables 6 and 7 where it has achieved a good performance in recovering the underlying (known) clustering structure. Hence, pF has been also used in the application to select the λ value maximizing (22).

5. Simulation Study

5.1. Design

In order to evaluate the performance of the ROOTCLUS model and the algorithm proposed in Sects. 3 and 4, an extensive simulation study has been carried on artificial data.

A number of three-way datasets have been generated by the true underlying model (2) by setting $H = 10$ (subjects), $N = 12, 24$ (objects), $J_{\text{true}} = 2, 3$ (clusters), $\%N_P = 50, 80$ (percentage of objects allocated to the Root partition). Then, each free-error similarity matrix $\mathbf{S}_h^*$ ($h = 1, \ldots, H$) has been built and perturbed as follows

$$\begin{aligned}\mathbf{S}_h &= \left(\mathbf{PWP}' + \left(\mathbf{P} + \mathbf{M}_h\right)\mathbf{V}_h\left(\mathbf{P} + \mathbf{M}_h\right)' + c_h\mathbf{1}_N\mathbf{1}_{N'}\right) + \delta\mathbf{E}_h \\ &= \left(\mathbf{S}_h^* + c_h\mathbf{1}_N\mathbf{1}_N'\right) + \delta\mathbf{E}_h\end{aligned}$$

where

- $\mathbf{P}$ is a random incomplete membership matrix having a percentage of nonzero rows equal to $\%N_P$ and generated with clusters of approximately equal sizes;
- $\mathbf{M}_h$ ($h = 1, \ldots, H$) are random incomplete membership matrices complementary to $\mathbf{P}$, having a percentage of zero rows equal to $\%N_P$ and the objects are randomly assigned to one out of the J_{true} Individual clusters with probability $1/J_{\text{true}}$;
- $\mathbf{W} = diag(4, 8)$ for $J_{\text{true}} = 2$ and $\mathbf{W} = diag(4, 8, 12)$ for $J_{\text{true}} = 3$;
- $\mathbf{V}_h = 2\mathbf{W}$ ($h = 1, \ldots, H$);
- $\mathbf{E}_h$ ($h = 1, \ldots, H$) are symmetric matrices of random noise generated by independently drawing their upper triangular entries from a standard normal distribution; then, matrices $\mathbf{E}_h$ have been rescaled so that their joint sum of squares $\sum_{h=1}^{H} \|\mathbf{E}_h\|^2$ was equal to $\sum_{h=1}^{H} \|\mathbf{S}_h^*\|^2$;
- δ has been set to 0.25, 0.5, 1 to allow for different error levels (Low, Medium, High) corresponding to 25%, 50%, 100% of noise-to-data ratio $\left(\sum_{h=1}^{H} \|\delta\mathbf{E}_h\|^2 \setminus \sum_{h=1}^{H} \|\mathbf{S}_h^*\|^2\right)$, respectively;
- c_h ($h = 1, \ldots, H$) have been chosen to ensure the nonnegativity of the final similarities.

For each cell of the experimental design, 100 datasets have been generated for a total of 2 (number of objects) $\times$ 2 (number of clusters) $\times$ 2 (% of objects in the Root partition) $\times$ 3 (error levels)= 2400 datasets.

The ROOTCLUS model has been fitted for a maximum number of clusters $J = 5$ and different values of λ (0–0.6 with increments of 0.1). Moreover, in order to prevent from falling into local optima, for any experimental cell and value of λ, the best solution in terms of *relative loss* (ESS/SS) has been retained starting from 100 different random solutions, so that the algorithm run 1,680,000 times in total.

Note that the simulation study was carried out on a personal computer with Intel(R) Core i7-7700 CPU 3.60 GHz processor and 16 GB of RAM.

The performance of ROOTCLUS according to different standpoints is assessed by computing several measures. Specifically, for each cell of the experimental design and for any value of λ the following measures have been computed by averaging over the 100 datasets:

- $\%P$: percentage of objects allocated to the Root optimal partition $\hat{\mathbf{P}}$; i.e., $100 \sum_{ij} \hat{\mathbf{p}}_{ij}/N$;
- MK: Kappa coefficient (Cohen 1960), to measure the similarity between true $\mathbf{P}$ and fitted $\hat{\mathbf{P}}$ Root partitions.[4] To take the permutational freedom of the (order of the) clusters into account, we defined the MK-statistic as the maximum value of the Kappa coefficient over all column permutations of the fitted $\hat{\mathbf{P}}$. As such, MK-statistic is the proportion of entries of the true $\mathbf{P}$, adjusted for chance, that have been recovered correctly; its maximum value is 1 meaning perfect recovery;
- $\%(MK = 1)$: percentage of successes in recovering the true partition $\mathbf{P}$, i.e., % of times where $MK = 1$;
- ARI: Adjusted Rand Index (Hubert and Arabie 1985) between true $\left(\mathbf{P} + \mathbf{M}_h\right)$ and fitted $\left(\hat{\mathbf{P}} + \hat{\mathbf{M}}_h\right)$ complete partitions averaged over the H subjects; it takes its maximum value equals to 1 when the two partitions are coincident;
- $\%(ARI = 1)$: percentage of successes in recovering the true partitions $\left(\mathbf{P} + \mathbf{M}_h\right)$, i.e., % of times where $ARI = 1$;
- $L2$: L2-distance between true $\mathbf{c}$ and fitted $\hat{\mathbf{c}}$ vectors of constants to evaluate the accuracy of the estimates;
- VAF: Variance Accounted For index (Hubert et al. 2006) between true and fitted proximity matrices;
- It: number of iterations before convergence;
- $Time$: time per run (in seconds);
- $\#sc$: number of non-empty clusters in the starting Root partition;
- $\#fc$: number of non-empty clusters in the final optimal Root partition.

The simulation results are displayed in Tables 6 and 7 where the average measures of performance are reported across the λ values for the two scenarios with $N = 12$ and $N = 24$, respectively.

5.2. *Results: Goodness of Recovery*

In order to investigate the importance of the manipulated factors of the simulation study and their interactions in determining the recovery performance, we performed an ANOVA with the ARI as dependent variable and the data-manipulated factors (N, $\%N_P$, J_{true} and *Error level*)

[4]The Kappa coefficient (KC) between two binary matrices is equal to the proportion of agreement between the two matrices (i.e., the proportion of the corresponding cells having the same values), corrected for chance (Wilderjans et al. 2012):

$$KC = \frac{(p_{00} + p_{11}) - (p_{0.}p_{.0} + p_{1.}p_{.1})}{1 - (p_{0.}p_{.0} + p_{1.}p_{.1})},$$

with p_{00} (p_{11}) being the proportion of corresponding cells that both are zero (one) and $p_{0.}$ and $p_{1.}$ ($p_{.0}$ and $p_{.1}$) the marginal proportion of zero- and one-cells for the first (second) matrix. Note that $p_{00} + p_{11}$ equals the (uncorrected) proportion of corresponding cells that have the same value.

TABLE 6.
Simulation study results ($N = 12$ and $H = 10$)

J_{true}	$\%N_P$	Error level	λ	*%P*	*MK*	*%(MK=1)*	*ARI*	*%(ARI=1)*	*L2*	*VAF*	*lt*	*Time*	*#sc*	*#fc*	*% pF**	*MK (pF*)*[a]
2	80	25%	0	82.92	0.9969	95	0.9990	97	0.0054	0.7228	17.82	0.0047	4.20	2.00	89	1
			0.1	83.33	1.0000	100	0.9993	98	0.0054	0.7227	16.96	0.0052	4.20	2.00	95	1
			0.2	83.33	1.0000	100	0.9993	98	0.0054	0.7227	16.81	0.0050	4.21	2.00	95	1
			0.3	83.50	0.9988	98	0.9980	96	0.0056	0.7205	16.85	0.0046	4.20	2.00	95	0.9994
			0.4	83.58	0.9983	97	0.9970	95	0.0058	0.7192	16.92	0.0043	4.20	2.00	94	0.9994
			0.5	84.08	0.9948	91	0.9857	87	0.0075	0.7094	16.49	0.0041	4.21	2.04	86	1
			0.6	85.33	0.9862	78	0.9648	75	0.0115	0.6903	15.70	0.0040	4.21	2.11	80	0.9958
		50%	0	77.75	0.9560	62	0.9863	70	0.0055	0.5874	17.24	0.0046	4.21	2.00	23	1
			0.1	83.33	1.0000	100	0.9926	80	0.0054	0.5860	16.66	0.0052	4.19	2.00	40	1
			0.2	83.42	0.9994	99	0.9913	78	0.0055	0.5849	16.73	0.0049	4.20	2.00	41	0.9986
			0.3	83.58	0.9983	97	0.9876	74	0.0059	0.5825	16.63	0.0044	4.20	2.01	40	0.9971
			0.4	85.33	0.9862	76	0.9531	52	0.0087	0.5578	15.94	0.0041	4.20	2.12	46	0.9875
			0.5	87.50	0.9714	56	0.9091	37	0.0122	0.5290	14.19	0.0042	4.20	2.26	55	0.9721
			0.6	90.25	0.9530	38	0.8548	21	0.0173	0.4942	13.44	0.0040	4.20	2.43	68	0.9496
		100%	0	61.33	0.8048	7	0.8928	4	0.0060	0.4578	15.82	0.0044	4.16	2.05	0	–
			0.1	83.17	0.9988	98	0.9618	29	0.0057	0.4495	16.12	0.0042	4.16	2.00	0	–
			0.2	83.75	0.9947	94	0.9525	26	0.0062	0.4447	15.32	0.0043	4.18	2.03	0	–
			0.3	86.17	0.9804	67	0.9106	17	0.0076	0.4239	14.23	0.0042	4.16	2.15	2	0.9156
			0.4	90.67	0.9501	33	0.8408	7	0.0103	0.3852	12.50	0.0038	4.16	2.36	25	0.8974
			0.5	94.83	0.9215	14	0.7673	2	0.0123	0.3511	11.11	0.0039	4.16	2.60	61	0.8968
			0.6	98.08	0.8989	0	0.7252	0	0.0136	0.3271	10.65	0.0038	4.17	2.71	90	0.8966

TABLE 6.
continued

J_{true}	$\%N_P$	Error level	λ	*%P*	*MK*	*%(MK=1)*	*ARI*	*%(ARI=1)*	*L2*	*VAF*	*lt*	*Time*	*#sc*	*#fc*	*% pF**	*MK(pF*)*[a]
2	50	25%	0	50.00	1.0000	100	1.0000	100	0.0059	0.7143	14.78	0.0044	3.90	2.00	99	1
			0.1	50.08	0.9991	99	1.0000	100	0.0059	0.7142	16.66	0.0052	3.90	2.00	100	0.9991
			0.2	51.58	0.9836	81	0.9910	82	0.0073	0.7051	17.46	0.0049	3.89	2.00	83	0.9979
			0.3	57.67	0.9226	25	0.8940	25	0.0223	0.6364	16.95	0.0045	3.90	2.25	26	0.9967
			0.4	66.50	0.8428	1	0.7212	1	0.0655	0.5223	13.76	0.0041	3.91	2.82	2	0.9569
			0.5	74.83	0.7768	0	0.5722	0	0.1126	0.4370	11.59	0.0039	3.90	3.36	0	–
			0.6	82.50	0.7216	0	0.4462	0	0.1550	0.3712	9.16	0.0039	3.88	3.82	0	–
		50%	0	49.33	0.9920	92	0.9967	91	0.0059	0.5785	14.94	0.0045	4.18	2.00	2	1
			0.1	50.83	0.9914	90	0.9914	83	0.0062	0.5757	17.26	0.0050	4.16	2.00	3	0.9713
			0.2	59.58	0.9036	10	0.8509	10	0.0161	0.5112	16.15	0.0047	4.17	2.32	3	0.9713
			0.3	71.08	0.8057	0	0.6262	0	0.0441	0.4135	11.97	0.0043	4.16	3.12	1	0.9138
			0.4	82.67	0.7205	0	0.4304	0	0.0778	0.3326	9.51	0.0037	4.18	3.90	10	0.6778
			0.5	88.58	0.6820	0	0.3522	0	0.0924	0.3004	8.44	0.0036	4.16	4.31	33	0.6616
			0.6	93.08	0.6549	0	0.3048	0	0.1007	0.2784	7.80	0.0037	4.16	4.34	65	0.6430
		100%	0	45.58	0.9333	58	0.9154	20	0.0064	0.4482	14.29	0.0041	4.23	2.05	0	–
			0.1	57.42	0.9238	16	0.7930	2	0.0102	0.4226	14.69	0.0045	4.23	2.43	0	–
			0.2	72.50	0.7831	0	0.5633	0	0.0242	0.3452	11.67	0.0038	4.22	3.26	0	–
			0.3	84.17	0.6992	0	0.3820	0	0.0400	0.2888	9.10	0.0038	4.23	4.02	0	–
			0.4	91.17	0.6521	0	0.3107	0	0.0478	0.2586	7.84	0.0037	4.22	4.40	18	0.6212
			0.5	95.83	0.6322	0	0.2831	0	0.0499	0.2415	7.80	0.0033	4.22	4.37	60	0.6149
			0.6	98.25	0.6185	0	0.2666	0	0.0515	0.2323	7.78	0.0034	4.22	4.37	89	0.6176

TABLE 6.
continued

J_{true}	$\%N_P$	Error level	λ	*%P*	*MK*	*%(MK=1)*	*ARI*	*%(ARI=1)*	*L2*	*VAF*	*lt*	*Time*	*#sc*	*#fc*	*% pF**	*MK (pF*)*[a]
3	80	25%	0	78.83	0.9632	72	0.9855	76	0.0037	0.7665	13.26	0.0039	4.02	3.00	71	1
			0.1	83.33	1.0000	100	0.9984	94	0.0036	0.7682	11.88	0.0045	4.00	3.00	99	1
			0.2	83.33	1.0000	100	0.9980	92	0.0036	0.7681	11.92	0.0043	4.01	3.00	97	1
			0.3	83.33	1.0000	100	0.9964	89	0.0037	0.7676	11.95	0.0042	4.02	3.00	93	1
			0.4	83.33	1.0000	100	0.9912	80	0.0039	0.7652	12.02	0.0038	4.01	3.00	83	1
			0.5	83.42	0.9994	99	0.9808	63	0.0046	0.7588	11.89	0.0035	4.01	3.00	66	1
			0.6	83.50	0.9988	98	0.9750	58	0.0051	0.7549	11.78	0.0035	4.01	3.01	60	1
		50%	0	65.83	0.8566	13	0.9254	8	0.0039	0.6394	12.32	0.0037	4.18	3.00	7	1
			0.1	83.17	0.9988	98	0.9879	61	0.0036	0.6377	11.22	0.0041	4.17	3.00	79	1
			0.2	83.33	1.0000	100	0.9859	55	0.0037	0.6367	11.25	0.0038	4.17	3.00	72	1
			0.3	83.42	0.9994	99	0.9759	40	0.0039	0.6331	11.05	0.0035	4.18	3.00	58	1
			0.4	83.42	0.9994	99	0.9667	36	0.0042	0.6288	11.09	0.0035	4.17	3.00	52	0.9989
			0.5	83.83	0.9965	94	0.9478	21	0.0047	0.6163	10.71	0.0034	4.17	3.00	35	0.9967
			0.6	84.67	0.9908	84	0.9244	11	0.0057	0.5995	10.44	0.0032	4.18	3.09	25	0.9862
		100%	0	42.08	0.6163	0	0.7667	0	0.0041	0.5075	10.86	0.0035	4.27	3.02	0	–
			0.1	82.00	0.9900	84	0.9480	11	0.0038	0.4966	10.67	0.0035	4.28	3.17	12	1
			0.2	83.25	0.9982	98	0.9411	6	0.0041	0.4904	10.48	0.0033	4.29	3.84	13	1
			0.3	83.75	0.9959	94	0.9117	3	0.0046	0.4779	9.98	0.0034	4.29	4.34	8	0.9784
			0.4	85.83	0.9815	70	0.8777	1	0.0055	0.4543	9.43	0.0034	4.27	4.69	17	0.9493
			0.5	88.50	0.9597	41	0.8439	1	0.0063	0.4261	8.55	0.0033	4.28	4.76	41	0.9385
			0.6	90.75	0.9421	20	0.8139	0	0.0071	0.4032	8.57	0.0033	4.28	4.83	43	0.9311

TABLE 6.
continued

J_{true}	$\%N_P$	Error level	λ	*%P*	*MK*	*%(MK=1)*	*ARI*	*%(ARI=1)*	*L2*	*VAF*	*lt*	*Time*	*#sc*	*#fc*	*% pF**	*MK(pF*)*[a]
3	50	25%	0	49.92	0.9990	99	0.9982	96	0.0042	0.7628	10.25	0.0036	4.31	3.00	98	1
			0.1	50.00	1.0000	100	0.9993	98	0.0041	0.7635	11.25	0.0043	4.33	3.00	100	1
			0.2	50.00	1.0000	100	0.9993	98	0.0041	0.7635	11.41	0.0043	4.33	3.00	100	1
			0.3	51.08	0.9888	87	0.9854	79	0.0050	0.7530	11.60	0.0042	4.32	3.02	80	1
			0.4	55.08	0.9476	41	0.9261	33	0.0089	0.7036	11.12	0.0042	4.32	3.18	33	1
			0.5	60.25	0.8968	2	0.8402	2	0.0165	0.6334	10.88	0.0039	4.32	3.54	2	1
			0.6	64.67	0.8572	0	0.7664	0	0.0269	0.5759	10.69	0.0037	4.32	3.71	0	–
		50%	0	47.92	0.9690	81	0.9767	51	0.0043	0.6317	10.16	0.0036	4.34	3.00	54	1
			0.1	50.00	1.0000	100	0.9901	65	0.0041	0.6348	11.37	0.0043	4.36	3.00	75	1
			0.2	52.33	0.9759	72	0.9514	39	0.0054	0.6158	11.59	0.0040	4.36	3.09	48	0.9982
			0.3	59.92	0.9000	4	0.8185	2	0.0103	0.5412	10.87	0.0039	4.35	3.48	3	0.9713
			0.4	66.83	0.8355	0	0.6967	0	0.0185	0.4718	9.64	0.0037	4.35	3.96	1	0.8387
			0.5	73.08	0.7870	0	0.5919	0	0.0274	0.4144	9.03	0.0035	4.35	4.33	3	0.7143
			0.6	79.00	0.7422	0	0.5041	0	0.0367	0.3656	7.85	0.0036	4.35	4.51	17	0.6852
		100%	0	38.67	0.8399	15	0.8338	0	0.0048	0.4912	10.35	0.0035	4.27	3.02	0	–
			0.1	52.25	0.9713	71	0.8687	3	0.0049	0.4899	10.96	0.0039	4.28	3.17	0	–
			0.2	64.08	0.8576	1	0.6785	0	0.0094	0.4143	9.25	0.0037	4.29	3.84	0	–
			0.3	72.83	0.7842	0	0.5415	0	0.0149	0.3573	8.54	0.0035	4.29	4.34	0	–
			0.4	80.92	0.7174	0	0.4307	0	0.0211	0.3072	7.30	0.0034	4.27	4.69	10	0.6962
			0.5	85.83	0.6795	0	0.3653	0	0.0245	0.2801	7.12	0.0033	4.28	4.76	31	0.6529
			0.6	90.17	0.6597	0	0.3327	0	0.0266	0.2586	7.15	0.0034	4.28	4.83	67	0.6529

[a]No value for *MK*(*pF**) is reported whenever *pF** never assumes its maximum for a given λ (*%pF**=0).

TABLE 7.
Simulation study results ($N = 24$ and $H = 10$)

J_{true}	$\%N_P$	Error level	λ	*%P*	*MK*	*%(MK=1)*	*ARI*	*%(ARI=1)*	*L2*	*VAF*	*lt*	*Time*	*#sc*	*#fc*	*% pF**	*MK (pF*)*[a]
2	80	25%	0	79.17	1.0000	100	1.0000	100	0.0010	0.6983	21.80	0.0160	4.99	2.00	100	1
			0.1	79.17	1.0000	100	1.0000	100	0.0010	0.6983	17.29	0.0152	4.99	2.00	100	1
			0.2	79.17	1.0000	100	1.0000	100	0.0010	0.6983	17.82	0.0156	4.99	2.00	100	1
			0.3	79.17	1.0000	100	1.0000	100	0.0010	0.6983	17.85	0.0156	4.99	2.00	100	1
			0.4	79.17	1.0000	100	1.0000	100	0.0010	0.6983	18.34	0.0148	4.99	2.00	100	1
			0.5	79.17	1.0000	100	1.0000	100	0.0010	0.6983	19.40	0.0154	4.99	2.00	100	1
			0.6	79.17	1.0000	100	1.0000	100	0.0010	0.6983	20.64	0.0149	4.99	2.00	100	1
		50%	0	78.88	0.9977	94	0.9995	97	0.0010	0.5532	21.03	0.0138	4.98	2.00	75	1
			0.1	79.17	1.0000	100	0.9995	97	0.0010	0.5532	17.41	0.0151	4.98	2.00	79	1
			0.2	79.17	1.0000	100	0.9995	97	0.0010	0.5532	17.48	0.0156	4.98	2.00	79	1
			0.3	79.17	1.0000	100	0.9995	97	0.0010	0.5532	18.57	0.0161	4.98	2.00	79	1
			0.4	79.21	0.9997	99	0.9995	97	0.0010	0.5531	21.07	0.0153	4.98	2.00	81	0.9996
			0.5	79.42	0.9982	94	0.9985	92	0.0010	0.5521	22.36	0.0142	4.98	2.00	86	0.9979
			0.6	81.54	0.9833	74	0.9567	72	0.0032	0.5338	21.66	0.0124	4.98	2.23	94	0.9848
		100%	0	77.50	0.9870	71	0.9937	65	0.0010	0.4116	19.70	0.0118	4.95	2.00	2	1
			0.1	79.17	1.0000	100	0.9943	69	0.0010	0.4114	17.57	0.0159	4.96	2.00	2	1
			0.2	79.17	1.0000	100	0.9945	70	0.0010	0.4114	18.71	0.0159	4.96	2.00	2	1
			0.3	79.71	0.9961	89	0.9887	65	0.0011	0.4095	20.57	0.0148	4.96	2.03	2	1
			0.4	82.50	0.9766	67	0.9332	48	0.0026	0.3945	20.67	0.0140	4.96	2.41	6	0.9147
			0.5	89.54	0.9277	17	0.7969	14	0.0063	0.3552	16.59	0.0130	4.95	3.29	25	0.8838
			0.6	95.92	0.8851	2	0.6922	2	0.0091	0.3261	12.79	0.0122	4.96	3.82	92	0.8827

TABLE 7.
continued

J_{true}	$\%N_P$	Error level	λ	$\%P$	MK	$\%(MK=1)$	ARI	$\%(ARI=1)$	$L2$	VAF	lt	$Time$	$\#sc$	$\#fc$	$\%\ pF^*$	$MK(pF^*)$[a]
2	50	25%	0	50.00	1.0000	100	1.0000	100	0.0010	0.6973	12.92	0.0122	4.95	2.00	99	1
			0.1	50.00	1.0000	100	1.0000	100	0.0010	0.6973	15.92	0.0162	4.95	2.00	99	1
			0.2	50.13	0.9987	97	0.9995	97	0.0010	0.6967	16.39	0.0169	4.95	2.00	98	0.9995
			0.3	50.88	0.9906	79	0.9962	79	0.0011	0.6918	17.45	0.0179	4.95	2.00	80	0.9994
			0.4	52.71	0.9711	39	0.9820	39	0.0018	0.6755	18.40	0.0191	4.95	2.00	40	0.9989
			0.5	54.79	0.9496	14	0.9612	14	0.0025	0.6538	18.89	0.0188	4.95	2.04	15	0.9970
			0.6	58.00	0.9177	1	0.9117	1	0.0068	0.6145	18.98	0.0186	4.95	2.27	1	1
		50%	0	50.00	1.0000	100	1.0000	100	0.0010	0.5528	12.90	0.0126	4.96	2.00	28	1
			0.1	50.04	0.9996	99	0.9998	99	0.0010	0.5527	16.23	0.0179	4.96	2.00	29	0.9985
			0.2	51.29	0.9862	70	0.9927	70	0.0011	0.5464	17.67	0.0201	4.96	2.00	36	0.9888
			0.3	54.21	0.9556	19	0.9639	19	0.0018	0.5254	19.82	0.0219	4.96	2.06	22	0.9721
			0.4	59.71	0.9014	0	0.8654	0	0.0063	0.4779	18.76	0.0191	4.96	2.64	6	0.8953
			0.5	67.25	0.8353	0	0.7142	0	0.0188	0.4155	15.93	0.0184	4.96	3.49	11	0.8622
			0.6	76.54	0.7631	0	0.5457	0	0.0371	0.3481	12.85	0.0166	4.96	4.23	53	0.7343
		100%	0	49.42	0.9932	89	0.9959	79	0.0010	0.4124	13.46	0.0137	4.97	2.00	0	–
			0.1	50.79	0.9915	81	0.9914	63	0.0010	0.4102	17.88	0.0215	4.97	2.00	0	–
			0.2	58.21	0.9157	1	0.8507	1	0.0034	0.3758	17.32	0.0220	4.98	2.80	0	–
			0.3	69.25	0.8188	0	0.6348	0	0.0119	0.3198	15.26	0.0187	4.97	3.94	0	–
			0.4	81.50	0.7281	0	0.4557	0	0.0227	0.2675	12.15	0.0160	4.97	4.70	0	–
			0.5	90.50	0.6705	0	0.3495	0	0.0296	0.2362	9.99	0.0155	4.97	4.89	24	0.6310
			0.6	96.21	0.6367	0	0.2907	0	0.0328	0.2193	9.03	0.0146	4.97	4.96	92	0.6347

TABLE 7.
continued

J_{true}	$\%N_P$	Error level	λ	$\%P$	MK	$\%(MK=1)$	ARI	$\%(ARI=1)$	$L2$	VAF	lt	$Time$	$\#sc$	$\#fc$	$\%\ pF^*$	$MK(pF^*)$[a]
3	80	25%	0	77.63	0.9864	92	0.9908	94	0.0007	0.7476	17.45	0.0142	4.95	2.98	91	1
			0.1	79.17	1.0000	100	1.0000	100	0.0007	0.7515	15.66	0.0169	4.95	3.00	100	1
			0.2	79.17	1.0000	100	1.0000	100	0.0007	0.7515	15.68	0.0176	4.95	3.00	100	1
			0.3	79.17	1.0000	100	1.0000	100	0.0007	0.7515	15.73	0.0177	4.95	3.00	100	1
			0.4	79.17	1.0000	100	1.0000	100	0.0007	0.7515	15.77	0.0174	4.95	3.00	100	1
			0.5	79.17	1.0000	100	1.0000	100	0.0007	0.7515	15.94	0.0174	4.95	3.00	100	1
			0.6	79.17	1.0000	100	1.0000	100	0.0007	0.7515	16.29	0.0174	4.95	3.00	100	1
		50%	0	74.29	0.9562	74	0.9847	69	0.0007	0.6096	17.02	0.0127	4.95	2.98	66	1
			0.1	79.17	1.0000	100	0.9983	88	0.0007	0.6143	15.54	0.0163	4.96	3.00	95	1
			0.2	79.17	1.0000	100	0.9982	88	0.0007	0.6143	15.90	0.0163	4.95	3.00	94	1
			0.3	79.17	1.0000	100	0.9983	89	0.0007	0.6143	16.00	0.0168	4.95	3.00	95	1
			0.4	79.21	0.9997	99	0.9978	89	0.0007	0.6140	16.40	0.0161	4.95	3.00	95	0.9997
			0.5	79.21	0.9997	99	0.9980	90	0.0007	0.6140	16.73	0.0157	4.95	3.00	96	0.9997
			0.6	79.25	0.9994	98	0.9979	90	0.0007	0.6139	17.19	0.0155	4.95	3.00	99	0.9994
		100%	0	65.67	0.8763	26	0.9613	12	0.0007	0.4641	16.83	0.0108	4.94	3.00	5	1
			0.1	79.13	0.9997	99	0.9854	35	0.0007	0.4670	15.77	0.0155	4.94	3.00	34	1
			0.2	79.17	1.0000	100	0.9862	34	0.0007	0.4669	16.87	0.0155	4.94	3.00	35	1
			0.3	79.33	0.9988	96	0.9847	30	0.0007	0.4662	17.00	0.0158	4.94	3.01	36	0.9983
			0.4	79.63	0.9966	89	0.9806	32	0.0008	0.4643	17.13	0.0142	4.94	3.03	38	0.9968
			0.5	80.75	0.9884	68	0.9597	22	0.0010	0.4559	16.79	0.0120	4.94	3.23	44	0.9896
			0.6	82.75	0.9740	39	0.9221	7	0.0015	0.4400	15.66	0.0117	4.93	3.57	76	0.9714

TABLE 7.
continued

J_{true}	$\%N_P$	Error level	λ	*%P*	*MK*	*%(MK=1)*	*ARI*	*%(ARI=1)*	*L2*	*VAF*	*lt*	*Time*	*#sc*	*#fc*	*% pF**	*MK(pF*)*[a]
3	50	25%	0	48.13	0.9689	93	0.9947	93	0.0009	0.7420	10.66	0.0098	4.75	3.00	93	1
			0.1	50.00	1.0000	100	1.0000	100	0.0007	0.7479	10.41	0.0113	4.74	3.00	100	1
			0.2	50.00	1.0000	100	1.0000	100	0.0007	0.7479	10.63	0.0119	4.74	3.00	100	1
			0.3	50.13	0.9987	97	0.9994	97	0.0007	0.7472	11.03	0.0118	4.75	3.00	97	1
			0.4	50.46	0.9951	89	0.9965	89	0.0008	0.7447	11.74	0.0126	4.74	3.01	89	1
			0.5	51.83	0.9804	58	0.9824	58	0.0011	0.7316	11.72	0.0129	4.75	3.06	58	1
			0.6	53.88	0.9590	24	0.9557	24	0.0018	0.7090	12.13	0.0136	4.75	3.29	24	1
		50%	0	47.33	0.9608	82	0.9930	85	0.0008	0.6051	10.51	0.0096	4.74	3.00	76	1
			0.1	50.00	1.0000	100	0.9998	99	0.0007	0.6101	10.57	0.0122	4.74	3.00	95	1
			0.2	50.17	0.9982	96	0.9990	95	0.0007	0.6094	11.22	0.0129	4.74	3.00	98	0.9986
			0.3	51.83	0.9804	59	0.9798	58	0.0009	0.5983	12.25	0.0136	4.74	3.10	64	0.9959
			0.4	55.58	0.9416	7	0.9261	7	0.0017	0.5664	12.37	0.0146	4.73	3.59	9	0.9855
			0.5	59.75	0.9008	0	0.8560	0	0.0037	0.5281	11.98	0.0138	4.73	4.01	0	–
			0.6	64.08	0.8615	0	0.7866	0	0.0066	0.4894	11.87	0.0140	4.73	4.40	0	–
		100%	0	39.92	0.8378	46	0.9682	29	0.0008	0.4511	10.31	0.0115	4.92	3.00	0	–
			0.1	50.33	0.9964	92	0.9902	58	0.0008	0.4616	11.99	0.0159	4.92	3.04	0	–
			0.2	54.33	0.9542	16	0.9184	16	0.0012	0.4425	12.77	0.0173	4.92	3.61	0	–
			0.3	61.50	0.8846	0	0.7955	0	0.0028	0.4010	12.30	0.0176	4.92	4.16	0	–
			0.4	68.04	0.8279	0	0.6939	0	0.0054	0.3615	11.80	0.0152	4.92	4.69	0	–
			0.5	74.83	0.7748	0	0.5946	0	0.0087	0.3237	10.83	0.0144	4.92	4.93	5	0.7671
			0.6	81.00	0.7308	0	0.4979	0	0.0119	0.2915	9.63	0.0134	4.92	4.99	95	0.7286

[a]No value for *MK*(*pF**) is reported whenever *pF** never assumes its maximum for a given λ (*%pF**=0).

TABLE 8.
Effect sizes (partial η^2) of the ARI for all main and first-order interaction effects of the manipulated factors. Large effect sizes in bold (partial $\eta^2 > 0.10$).

Effects	λ						
	0	0.1	0.2	0.3	0.4	0.5	0.6
N	**0.12**	0.09	0.09	0.09	0.10	0.09	0.07
Error level	**0.23**	**0.21**	**0.25**	**0.25**	**0.23**	**0.24**	**0.23**
$N \times$ *Error level*	**0.14**	**0.14**	0.07	0.03	0.01	0.00	0.01
$\%N_P$	0.01	0.03	**0.14**	**0.25**	**0.33**	**0.39**	**0.45**
$N \times \%N_P$	0.01	0.03	0.05	0.05	0.04	0.04	0.03
Error level $\times \%N_P$	0.00	0.06	**0.15**	**0.15**	0.10	0.06	0.03
J_{true}	0.05	0.00	0.01	0.02	0.03	0.05	0.06
$N \times J_{\text{true}}$	0.01	0.00	0.00	0.00	0.00	0.00	0.00
Error level $\times J_{\text{true}}$	0.03	0.00	0.00	0.00	0.00	0.00	0.00
$\%N_P \times J_{\text{true}}$	0.00	0.01	0.01	0.02	0.02	0.01	0.01

as independent variables. Effect sizes (partial η^2) for each of the main effects and first-order interactions are displayed in Table 8.

In Figs. 1 and 2, the mean plots with confidence intervals of the ARI LS-means from post-hoc analyses have been reported to relate the results back to the experiment and better highlight how the algorithm performs as the manipulated factors with large effect sizes vary. LS-means and 95% confidence limits are displayed for each level of the main effects and interactions with large effect sizes in bold in Table 8. From the multiple-comparison analysis, the adjusted pairwise differences and their significance levels have shown that some interactions have similar effects (the corresponding intervals are connected by arrowed lines in Fig. 2b, d), while the main effects are all significant (see Figs. 1, 2a, c).

Thus, in this analysis, taking into account only effects with partial η^2 larger than 0.10 as relevant, it emerges that the *Error level* has a sizeable main effect across all λ values (partial η^2 from 0.21 to 0.25): As expected and evident from Tables 6 and 7, the recovery performance decreases as the error becomes larger (Fig. 2a).

The $\%N_P$ has a sizeable main effect increasing with $\lambda > 0.1$: When the percentage of objects in the Root partition is smaller ($\%N_P = 50$), the decrease in recovery performance is more pronounced as λ increases (Tables 6, 7 and Fig. 2c). Such a main effect is qualified by an *Error level* by $\%N_P$ interaction (partial $\eta^2 = 0.15$) for $\lambda = 0.2, 0.3$, which actually correspond to the λs where the algorithm is mostly likely to return the optimal solution and is more pronounced for a large amount of error (Fig. 2d).

The number of objects N determines a main effect only for $\lambda = 0$ (partial $\eta^2 = 0.12$), and it is qualified by an interaction between N and *Error level* (partial $\eta^2 = 0.14$) for λ equals to 0 and 0.1. In fact, for the smaller number of objects it emerges that the optimal solution is less likely to be found for $\lambda = 0$ (Fig. 1) and, again, this is more pronounced as the error level increases (Fig. 2b).

Finally, note that the true number of clusters J_{true} and any interactions with it indicate no differences in effect size for any λ value.

The recovery performance is generally good even when the error is quite high (Fig. 2a): it is better in terms of recovery of the true partitions when the number of objects is larger regardless of the λ values (Fig. 2b).

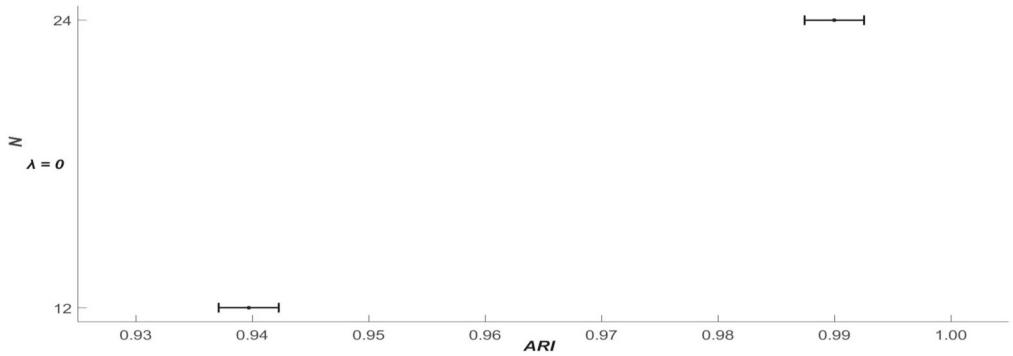

FIGURE 1.
Plot of ARI LS-means for number of objects N with 95% confidence limits.

In both settings, the recovery of the correct both Root and Individual classifications attains nearly comparable high values on average and the percentage of hitting the maximum agreement with the true Root partition ($MK = 1$) remains generally very high for at least one of the values of λ, except for an high level of error where it gets worse, especially for $\%N_P = 50$ than $\%N_P = 80$ (Tables 6, 7).

Probably, the high *Error level* combined with the presence of high heterogeneity of the subjects (only the 50% of the objects belonging to the common partition) makes the recovery of the true clustering structure more challenging.

Note that the results for $\lambda = 0$ are generally worse and probably affected by the tendency toward trivial solutions (Sect. 4.1.3).

Finally, it can be noted that regardless of the amount of error, it generally takes a larger amount of average time per run for those λ values where the correct classifications are identified, as expected and for $\%N_P = 50$ which is due to more steps required by the allocation step of the algorithm.

5.3. Results: Model Selection

In order to assess the performance of the algorithm in terms of model selection (i.e., determining the optimal number of clusters), the average number of non-empty clusters, out of 5 requested, in the starting and optimal Root partitions is reported in Tables 6 and 7. From the comparison, it clearly emerges that the number of starting non-empty clusters is generally larger than the optimal one for all settings.

This confirms that the number of the starting non-empty clusters ($\#sc$) does not affect the number of non-empty clusters one ends up ($\#fc$). Note that $\#fc$ is generally close to J_{true} for at least some λ values, depending on the amount of error. To evaluate how well the algorithm retrieves the correct clusters, the $\%(MK = 1)$ reports how often (in %) the perfect recovery of the true Root partition is achieved. For example, for the case ($J_{\text{true}} = 2$, $\%N_P = 50$, *Error level*$= 25\%$) in Table 6, the algorithm retrieves optimal solutions with two non-empty clusters ($\#fc = 2$) for $\lambda = 0, 0.1, 0.2$, whereas $\%(MK = 1)$ is equal to 100, 99, 81, respectively, which indicates how many of such optimal solutions into two clusters coincide with the true ones.

Note that it is not expected to find the correct clusters for all λs because of the penalty in the loss which forces toward larger and larger Root clusters as λ increases. What we expect is to find large percentages of correct recovery ($\%(MK = 1)$) for some λ values which are often obtained depending on the amount of error. For example, the worst of the best cases across λ is $\%(MK = 1) = 58$ corresponding to $\lambda = 0$ for the setting ($N = 12$, $J_{\text{true}} = 2$, $\%N_P =$

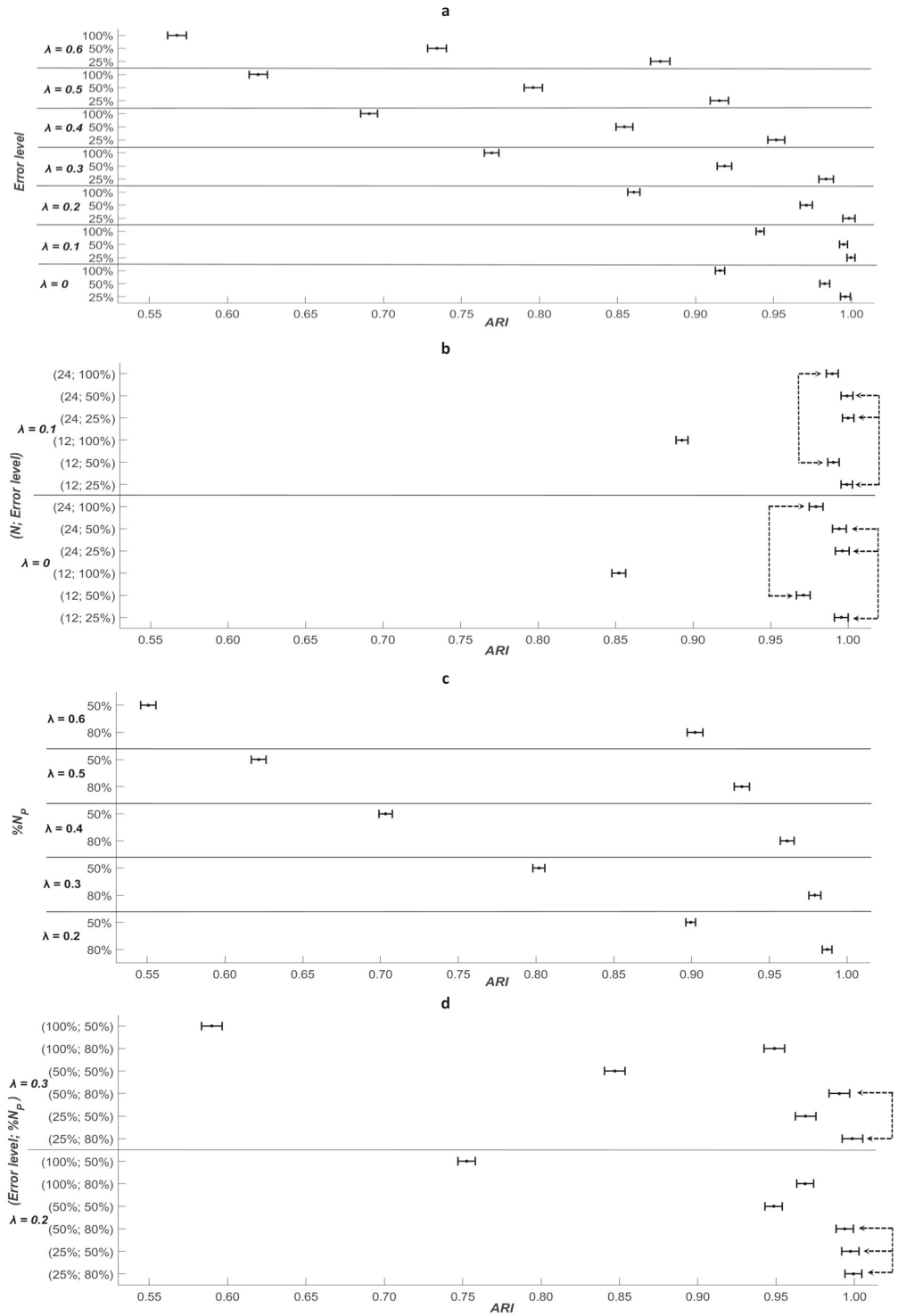

FIGURE 2.
Plot of ARI LS-means with 95% confidence limits for **a** *Error level*, **b** N by *Error level* interaction, **c** $\%N_P$ and **d** *Error level* by $\%N_P$ interaction (arrowed lines denote effects non-significantly different).

50, *Error level* = 100%). This confirms the capability of the algorithm to find the correct clusters for "at least" one λ value in different conditions and regardless of the starting solution.

Moreover, the simulation setup has been also used to assess the performance of the maximum pF index (22) across the λ values in selecting the most appropriate model. For any experimental cell and for any λ, Tables 6 and 7 report also:

- $\%pF^*$: percentage of times the pF index assumes its maximum (for the λ value in question);
- $MK(pF^*)$: average MK computed for the solutions indicated by pF^*.

The larger the $MK(pF^*)$ values, the better the ability of the pF index in recovering the underlying common clustering structure when it is unknown as in real applications, because the pF index manages to indicate lambda values giving good solutions in terms of recovery of the true Root partition.

As expected, the performance generally decreases as the noise level increases but it always succeeds in indicating a good solution on average except for the cases with high noise level combined with $\%N_P = 50$ where the model attains the worst fit as already observed.

5.4. *Number of Starts*

In order to analyze the stability of the solution and investigate on the sensitivity to local optima, additional results on the recovery performance are reported in Table 9 for the settings $H = 10$, $J_{\text{true}} = 3$, $N = 12, 24$, $\%N_P = 50, 80$ and *Error level* = 50%, 100%, being the data generated as described in Sect. 5.1 and the ROOTCLUS model fitted for a maximum number of clusters $J = 5$.

When the best solution is retained in an increasing number of random starts (10, 30, 50, 100), both the average ARI between true and fitted partitions and the % of times where $ARI = 1$ show on average a fairly good improvement across λ values, as expected. A good performance in terms of average ARI is already reached when the optimal solution is retained over a few random starts when it generally becomes stable. Nonetheless, the percentage of times where the correct partitions are perfectly recovered for *all* subjects needs a larger number of random starts for a more relevant improvement, especially for small datasets.

5.5. *Scalability*

A small simulation has been also carried out to evaluate the average runtime for larger data. In a setting where $J_{\text{true}} = 3$, $\%N_P = 50$ and *Error level* = 50%, ten datasets have been generated as described in Sect. 5.1 by varying the number of objects $N = 12, 24, 48$ and the number of subjects $H = 10, 20, 40$. As before, the model has been fitted for a maximum number of clusters $J = 5$, λ from 0 to 0.6 with increments of 0.1 and by retaining the best solution in 100 different random starts. The average total runtime (in seconds) to get the optimal solution is reported in Table 10.

Generally speaking, a large number of subjects are more burdensome than a large number of objects due to the larger number of the Individual partitions, as expected. Given the number of objects, the average runtime increases by about 4.7 times from $H = 10$ to $H = 20$ subjects and it increases by 3.5 times more for $H = 40$. On the other hand, given the number of subjects, as the number of objects goes from 12 to 24 and then from 24 to 48, the runtime goes up a factor of 3.1 and 4.9 on average, respectively.

Note that such a total runtime has been obtained by retaining the best solution in 100 random starts, by considering 7 values for λ and by searching for 5 possible non-empty clusters. In situations with larger data in terms of number of either objects or subjects, time can be possibly saved by reducing the increments of the λ values or the number of random starts.

TABLE 9.
Analysis of random starts ($H = 10$, $J_{\text{true}} = 3$)

Number of starts	*ARI* λ							Number of starts	%(*ARI*=1) λ						
	0	0.1	0.2	0.3	0.4	0.5	0.6		0	0.1	0.2	0.3	0.4	0.5	0.6
	$N = 12$ $\%N_P = 80$ *Error level*=50%								$N = 12$ $\%N_P = 80$ *Error level*=50%						
10	0.8655	0.9781	0.9627	0.9455	0.9156	0.8842	0.8600	10	7	45	32	16	9	5	2
30	0.9022	0.9859	0.9772	0.9662	0.9463	0.9222	0.8864	30	6	58	43	29	18	7	4
50	0.9191	0.9871	0.9811	0.9704	0.9552	0.9347	0.9119	50	8	60	48	35	25	16	9
100	0.9254	0.9879	0.9859	0.9759	0.9667	0.9478	0.9244	100	8	61	55	40	36	21	11
	$N = 12$ $\%N_P = 80$ *Error level*=100%								$N = 12$ $\%N_P = 80$ *Error level*=100%						
10	0.7106	0.9237	0.8982	0.8555	0.8250	0.7801	0.7527	10	0	4	2	0	0	0	0
30	0.7408	0.9420	0.9204	0.8892	0.8557	0.8108	0.7882	30	0	11	2	2	0	0	0
50	0.7550	0.9451	0.9269	0.9003	0.8649	0.8255	0.8048	50	0	10	2	2	1	0	0
100	0.7667	0.9480	0.9411	0.9117	0.8777	0.8439	0.8139	100	0	11	6	3	1	1	0
	$N = 12$ $\%N_P = 50$ *Error level*=50%								$N = 12$ $\%N_P = 50$ *Error level*=50%						
10	0.9164	0.9700	0.8651	0.7104	0.5935	0.5018	0.4279	10	27	44	9	0	0	0	0
30	0.9576	0.9867	0.9188	0.7704	0.6610	0.5510	0.4704	30	39	59	23	1	0	0	0
50	0.9731	0.9887	0.9362	0.7942	0.6794	0.5698	0.4846	50	47	63	33	1	0	0	0
100	0.9767	0.9901	0.9514	0.8185	0.6967	0.5919	0.5041	100	51	65	39	2	0	0	0
	$N = 12$ $\%N_P = 50$ *Error level*=100%								$N = 12$ $\%N_P = 50$ *Error level*=100%						
10	0.7414	0.7801	0.5881	0.4608	0.3529	0.3282	0.2964	10	0	0	0	0	0	0	0
30	0.8011	0.8361	0.6475	0.4982	0.3959	0.3461	0.3213	30	0	1	0	0	0	0	0
50	0.8211	0.8521	0.6565	0.5160	0.4108	0.3539	0.3316	50	0	2	0	0	0	0	0
100	0.8338	0.8687	0.6785	0.5415	0.4307	0.3653	0.3327	100	0	3	0	0	0	0	0

TABLE 9.
continued

Number of starts	*ARI* λ							Number of starts	%(*ARI*=1) λ						
	0	0.1	0.2	0.3	0.4	0.5	0.6		0	0.1	0.2	0.3	0.4	0.5	0.6
	$N = 24$ $\%N_P = 80$ *Error level*=50%								$N = 24$ $\%N_P = 80$ *Error level*=50%						
10	0.9496	0.9981	0.9979	0.9981	0.9979	0.9940	0.9909	10	47	86	87	87	86	77	73
30	0.9606	0.9981	0.9981	0.9982	0.9981	0.9967	0.9958	30	57	86	87	88	88	84	85
50	0.9749	0.9983	0.9982	0.9983	0.9980	0.9981	0.9961	50	63	88	88	89	89	90	87
100	0.9847	0.9983	0.9982	0.9983	0.9978	0.9980	0.9979	100	69	88	88	89	89	90	90
	$N = 24$ $\%N_P = 80$ *Error level*=100%								$N = 24$ $\%N_P = 80$ *Error level*=100%						
10	0.9144	0.9857	0.9856	0.9798	0.9539	0.9131	0.8504	10	3	36	30	23	20	9	2
30	0.9398	0.9853	0.9857	0.9838	0.9727	0.9434	0.8931	30	13	36	32	28	27	17	4
50	0.9491	0.9852	0.9860	0.9841	0.9757	0.9497	0.9116	50	12	35	33	29	28	17	6
100	0.9613	0.9854	0.9862	0.9847	0.9806	0.9597	0.9221	100	12	35	34	30	32	22	7
	$N = 24$ $\%N_P = 50$ *Error level*=50%								$N = 24$ $\%N_P = 50$ *Error level*=50%						
10	0.9183	0.9999	0.9986	0.9671	0.8924	0.8073	0.7375	10	39	99	94	50	3	0	0
30	0.9779	0.9998	0.9990	0.9774	0.9158	0.8325	0.7677	30	61	99	95	56	4	0	0
50	0.9865	0.9998	0.9990	0.9784	0.9211	0.8460	0.7730	50	73	99	95	56	5	0	0
100	0.9930	0.9998	0.9990	0.9798	0.9261	0.8560	0.7866	100	85	99	95	58	7	0	0
	$N = 24$ $\%N_P = 50$ *Error level*=100%								$N = 24$ $\%N_P = 50$ *Error level*=100%						
10	0.8877	0.9930	0.8870	0.7553	0.6405	0.5417	0.4440	10	11	57	7	0	0	0	0
30	0.9451	0.9923	0.9122	0.7769	0.6708	0.5655	0.4759	30	21	58	15	0	0	0	0
50	0.9527	0.9909	0.9158	0.7833	0.6856	0.5780	0.4874	50	22	58	15	0	0	0	0
100	0.9682	0.9902	0.9184	0.7955	0.6939	0.5946	0.4979	100	29	58	16	0	0	0	0

TABLE 10.
Average total runtime (in seconds) to get the optimal solution ($J_{\text{true}} = 3$, $\%N_P = 50$, *Error level* = 50%)

N	H		
	10	20	40
12	0.027	0.138	0.503
24	0.091	0.445	1.385
48	0.470	1.969	7.102

6. Illustrative Applications

6.1. Cola Data

We analyze the well-known proximity Cola data among all pairs of ten colas provided by ten subjects (Schiffman et al. 1981, pp. 33–34). In a sensory experiment, ten subjects (nonsmokers, aged 18–21 years) provided 45 dissimilarity judgments tasting ten different brands of cola: Diet Pepsi (DiP), RC Cola (RCC), Yukon (Yuk), Dr. Pepper (DrP), Shasta (Sha), Coca Cola (CoC), Diet Dr. Pepper (DDP), Tab (Tab), Pepsi Cola (PeC), Diet Rite (DiR). The dissimilarity judgements for each pair of colas were placed on a scale from 0 (representing "same") to 100 (representing "different").

In order to fit the ROOTCLUS model, the dissimilarity judgements have been converted into similarities by taking the complements to 100 of the original values.

The aim is here to investigate the existence of sensory perceptual heterogeneity among subjects using the proposed methodology. Actually, the ten subjects can show different perception of colas depending on two key elements: (1) Five subjects (A, D, E, F, I) have got one dominant allele on the human genome which affects their ability to taste a bitter compound called PTC (which is bitter to all of them but is tasteless to all the others), while the other five subjects (B, C, G, H, J) do not have this inheritable characteristic (Schiffman et al. 1981, p. 151); (2) Colas can be classified either diet (DiP, DDP, Tab, DiR) or non-diet (RCC, Yuk, DrP, Sha, CoC, PeC) and either cherry (DrP, DDP) or noncherry (DiP, RCC, Yuk, Sha, CoC, Tab, PeC, DiR).

Therefore, we wish to investigate the existence of common subsets of colas perceived as similar by all subjects and, at the same time, identify how each subject can differently link the remaining colas to such common clusters.

The best solution has been retained over 300 random starts. The ROOTCLUS model has been fitted to the off-diagonal entries of the ten similarity matrices for different values of λ (0–1, with increments of 0.1) and setting the maximum number of clusters to $J = 4$. The best solution results in two non-empty clusters for $\lambda = 0$ (unconstrained solution) and in three non-empty clusters for larger λ values.

From a first analysis of all the Root partitions, it is worth noticing that the Root clusters remain quite stable across the different values of λ (Table 11), identifying actually the "roots" of different types of cola: R_1 = regular noncherry colas, R_2 = cherry colas (either DrP or DDP) and R_3 = diet noncherry colas.

The model selection in terms of the choice of the best solutions across the λ values has been evaluated by inspecting the scree plot of both the relative loss and the pF index (Fig. 3).

The pF index attains its maximum for $\lambda \geq 0.6$, but also at $\lambda = 0$ it attains a local maximum: They identify the two extreme situations where either all colas but DDP belong to the common partition or only 5 out of 10 colas are assigned to the Root clusters.

In the first case ($\lambda \geq 0.6$), the (parsimonious) solution goes toward a unique partition and the heterogeneity of the subjects depends on how they differently taste the only diet cherry cola

TABLE 11.
Cola data: results from ROOTCLUS

λ	Relative loss	Root partition			
		R_1	R_2	R_3	R_4
0.0	0.1862	RCC, Sha, CoC, PeC	DDP	–	–
0.1, 0.2	0.2071	RCC, Sha, CoC, PeC	DDP	Tab	–
0.3, 0.4	0.2195	RCC, Sha, CoC, PeC	DrP	Tab, DiP	–
0.5	0.2322	RCC, Sha, CoC, PeC	DrP	Tab, DiP, DiR	–
0.6–1	0.2517	RCC, Sha, CoC, PeC, Yuk	DrP	Tab, DiP, DiR	–

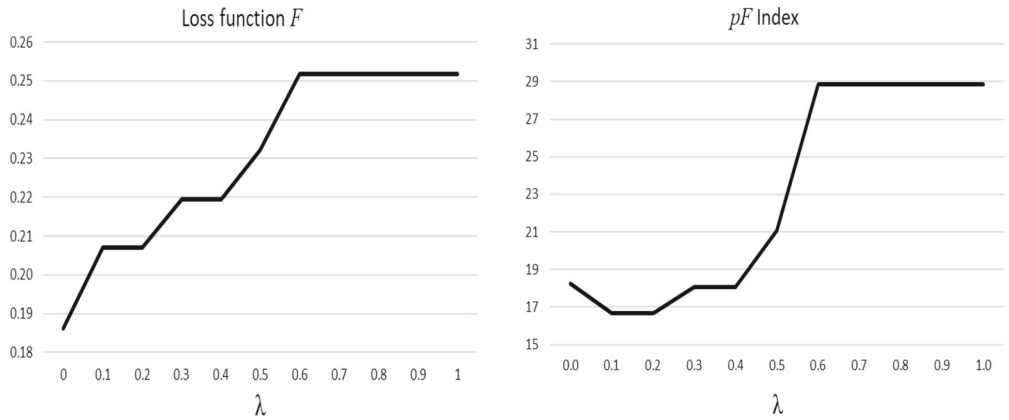

FIGURE 3.
Cola data: loss function F and pF index against λ values.

DDP (Table 12): the 5 non-PTC tasters add DDP to R_2 to form a cluster of *cherry colas* (Cluster 2), while the 5 PTC subjects put DDP with the diet noncherry ones in R_3 forming a cluster of *diet colas* (Cluster 3). Therefore, the PTC tasters recognize the DDP as diet, whereas this is not true for the non-PTC tasters: this implies that PTC tasters evaluate DDP more similar to the colas in R_3 (because it is diet), whereas such a resemblance is not perceived by the non-PTCs.

The optimal weights of the Root clusters are $\mathbf{w} = (31.57, -, 17.29)$:[5] the regular noncherry colas (R_1) tasted much more similar than diet ones (R_3) for all subjects with a baseline similarity about 1.8 times larger. Moreover, the weights of the Individual clusters in Fig. 4 help to better qualify such a result: For PTC subjects, the diet colas in R_3 are more similar to each other than to DDP. Because of the estimated weights, the similarity between Tab, DiP and DiR in R_3 is given by the baseline similarity $w_3 = 17.29$ for non-PTCs. Conversely, for PTCs the similarity between Tab, DiP and DiR in R_3 is larger. For example, subject E evaluates the similarity between the three colas in R_3 equal to $w_3 + v_{3E} = 17.29 + 33.19 = 50.48$, while DDP results less similar to the diet noncherry colas ($v_{3E} = 33.19$). This is due to the additional cherry flavor which characterizes the diet DDP. The same reasoning and interpretation can be done for Cluster 2 with the regular cherry DrP for PTC tasters, while it additionally includes the diet cherry DDP according to the non-PTCs: The latter perceive only the cherry flavor (and not the diet one) common to this two

[5]Note that w_2 is missing here because the Root cluster R_2 is a singleton and the diagonal entries of the similarity matrices are not fitted in this application.

TABLE 12.
Cola data: clusters for PTC (A, D, E, F, I) and non-PTC (B, C, G, H, J) tasters from ROOTCLUS ($\lambda = 0.6$)

Cluster 1 *Regular colas*		Cluster 2 *Cherry colas*		Cluster 3 *Diet colas*	
R_1 *Regular Noncherry colas*		R_2 *Regular Cherry colas*		R_3 *Diet Noncherry colas*	
RCC Sha CoC PeC Yuk		DrP		Tab DiP DiR	
PTC	Non-PTC	PTC	Non-PTC	PTC	Non-PTC
I_1	I_1	I_2	I_2 *Diet cherry cola*	I_3 *Diet cherry cola*	I_3
–	–	–	DDP	DDP	–

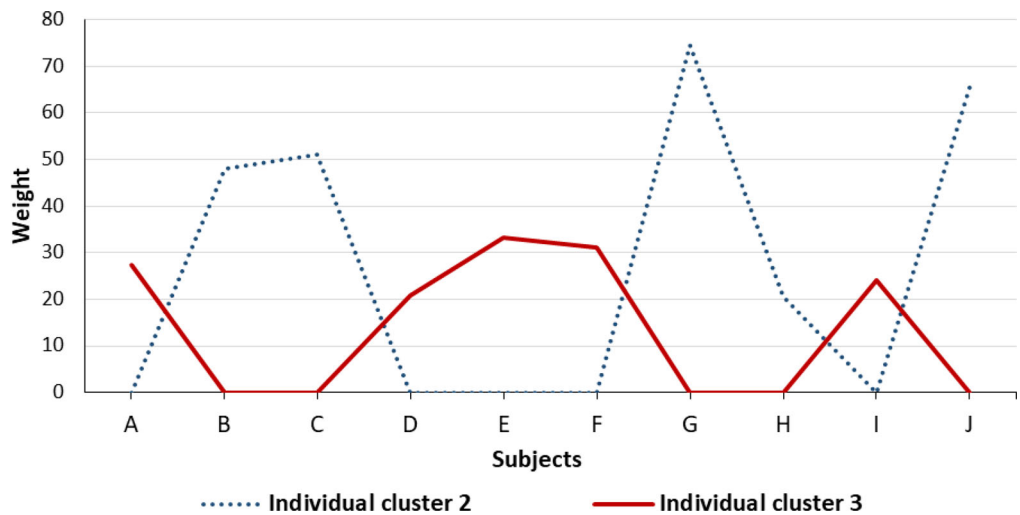

FIGURE 4.
Cola data: weights of the ten subjects for each Individual cluster ($\lambda = 0.6$). The weights for any Individual cluster 1 are not reported because they are 0 for all subjects.

colas and evaluate them very similar (large weights of the Individual clusters 2 for non-PTC tasters as evident from Fig. 4).

For the case $\lambda = 0$, the (most flexible) optimal solution provides two Root clusters (Table 11) including only five non-diet colas (4 noncherry in R_1 and 1 cherry in R_2), while the remaining five colas differentiate the Individual clusters (Table 13).

Hence, together with the Root clusters, the Individual partitions reflect the ability of the subjects to differently taste the colas because the PTC subjects actually distinguish non-diet (Cluster 1) vs diet (Cluster 2) colas, while non-PTC subjects mainly tend to separate noncherry (Cluster 1) from cherry (Cluster 2) colas. Therefore, this solution, as for $\lambda = 0.6$, reveals that the non-PTC subjects are not able to perceive the different tastes of the diet colas.

TABLE 13.
Cola data: Root and Individual clusters from ROOTCLUS ($\lambda = 0$)

	Cluster 1 R_1	Cluster 2 R_2
	RCC Sha CoC PeC	DDP
Subjects	I_1	I_2
PTC tasters A	Yuk	Tab, DiP, DiR, *DrP*
PTC tasters (D, E, F, I)	Yuk, *DrP*	Tab, DiP, DiR
Non-PTC tasters (B, C)	Tab, DiP, DiR	DrP, *Yuk*
Non-PTC tasters (G, H, J)	Tab, DiP, DiR, *Yuk*	DrP

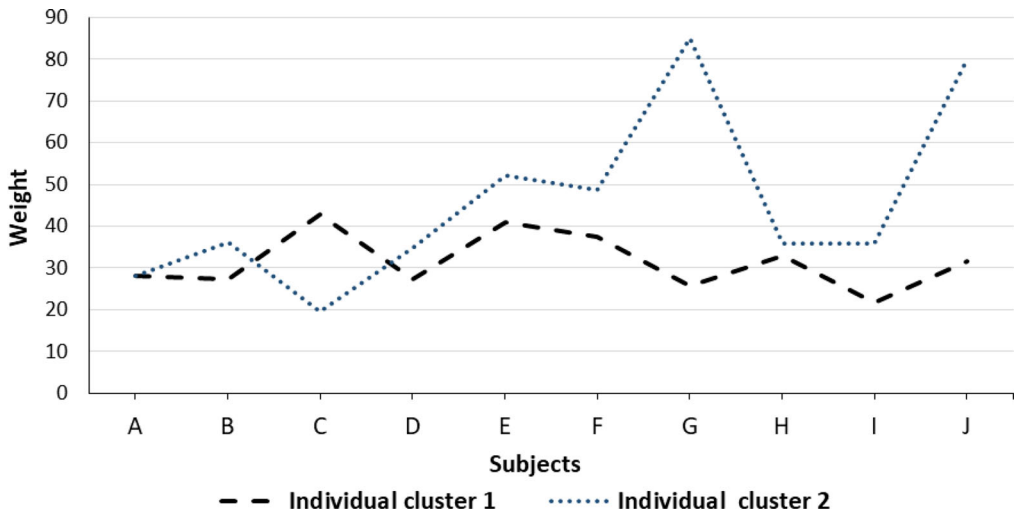

FIGURE 5.
Cola data: weights of the ten subjects for each Individual cluster ($\lambda = 0$).

The optimal weights of the Root clusters are $\mathbf{w} = (18.02, -)$, while the optimal weights of the Individual clusters are displayed in Fig. 5 where it is evident that colas within Individual cluster 2 are perceived more similar to each other than colas in the Individual cluster 1, except for subject C. Moreover, from the weights in Fig. 5 it emerges that for both subjects G and J the cherry cola DrP (in I_2) is the most similar to the diet cherry cola DDP (the only in R_2) because of the largest weights $v_{2G} = 85$ and $v_{2J} = 79.8$: This is due probably to the cherry taste of DrP and DDP which are indistinguishable to the two non-PTC tasters G and J.

The results from ROOTCLUS, interpretable in terms of the particular inheritable characteristic of the subjects, confirm the existence of the same perceptual heterogeneity among subjects (see, for instance, Schiffman et al. 1981; Wedel and DeSarbo 1998; Bocci and Vicari 2017).

In fact, the results from the ROOTCLUS model are consistent with the ones from INDSCAL (Schiffman et al. 1981, pp. 149–151) where for the PTC subjects the distinction between diet and

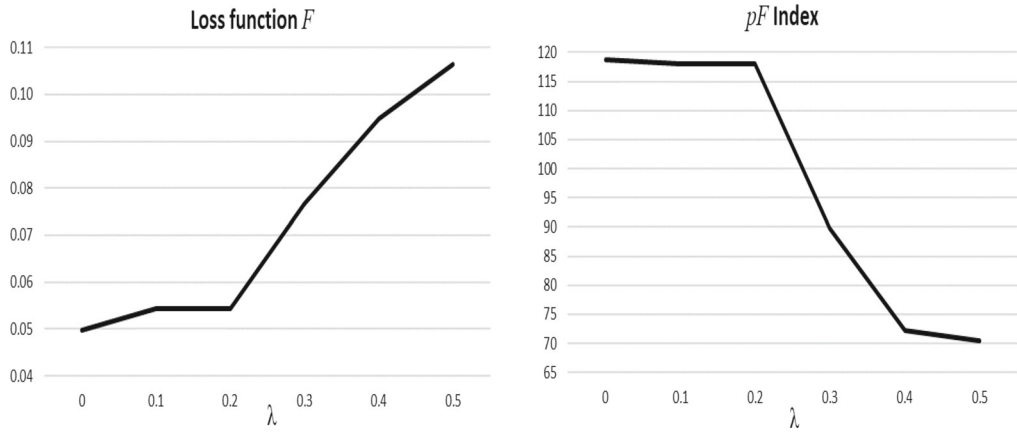

FIGURE 6.
Sport data: loss function F and pF index against λ values.

non-diet colas is more important than that between cherry and regular colas, while for non-PTCs the reverse is true.

Moreover, the ROOTCLUS solution is also consistent with the one from INDCLUS with $J = 2$ clusters of colas. The first cluster consists of seven colas (four regular noncherry colas—RCC, Sha, CoC, PeC—plus their diet versions—DiR, Tab, DiP), the second cluster is formed only by regular noncherry colas (RCC, Sha, CoC, PeC, Yuk), while the two cherry colas (DrP and DDP) are not assigned to any cluster. In INDCLUS, the differences in judging similarities are given by the subject weights: All PTC tasters (A, D, E, F, I) tend to weigh much more than the second cluster of regular colas, while the non-PTC subjects (except for H) do the opposite (see Bocci and Vicari 2017, for numerical details).

6.2. *Sport Data*

The Sport data already analyzed by gender in Sect. 2 have been fully investigated by fitting the ROOTCLUS model to the 13 student pairwise similarity matrices having entries equal to either ones or zeros whether two sports belong to the same cluster or not.

It was chosen to analyze these data by setting the maximum number of clusters to 7 and the algorithm run for λ values in [0; 0.5] with increments of 0.1 by retaining the best solution from 200 different starts.

In Fig. 6, the loss function F is plotted against the λ values together with the pF index (22). The ROOTCLUS loss function assumes (almost) the same values for $\lambda \leq 0.2$ before increasing quickly, the pF index attains its maximum for $\lambda = 0$ and stabilizes to a value close to the maximum at $\lambda = 0.1, 0.2$ before dropping down rapidly for larger λs.

Since the optimal Root clusters for $\lambda = 0$ and $\lambda = 0.1, 0.2$ are equal up to one sport, the latter solution is analyzed in detail because of its parsimony.

It is worthwhile noticing that three non-empty Root clusters are always found across λs and the number of sports allocated to the Root partitions varies from 6 to 9 in such solutions.

The solution for $\lambda = 0.1$ finds three non-empty clusters which accommodate in total the 50% of the sports: R_1 =(Volleyball, Water Polo, Rugby, Soccer), R_2 =(Horse-riding), R_3 =(Athletics, Artistic Gym).

The remaining sports are differently allocated by the 13 students as displayed in Fig. 7 where, for each of the seven sports not belonging to the Root clusters, the bars show the different

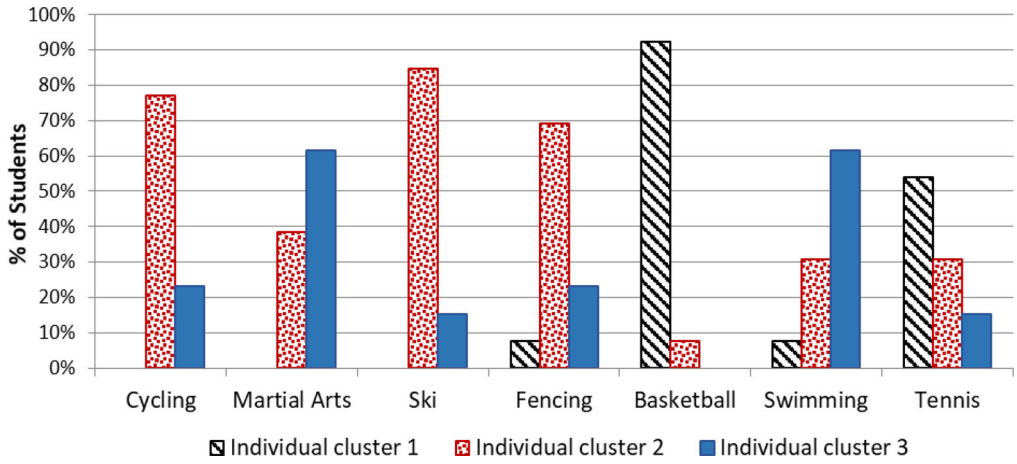

FIGURE 7.
Sport data: sports in Individual clusters from ROOTCLUS (% of students).

compositions of the Individual clusters. The heights of the bars (in different patterns) indicate the percentage of students who have put each sport into each Individual cluster.

Thus, it is evident that Basketball has been considered very similar to the *Team Sports* in R_1 by almost all the students (92%), while only half of them have added Tennis to the sports in R_1 to form a cluster interpretable as *Ball Sports*. The majority of students have put Ski, Cycling and Fencing with Horse-riding to form a cluster of sports requiring *special equipment* and a minority of students have also added Martial Arts, Swimming and Tennis with them. Finally, the third cluster of the *Multidisciplinary Sports* is formed mainly by adding Martial Arts and Swimming to Athletics and Artistic Gym in R_3 (only a few students include here also Fencing and the other sports).

Such partitionings reflect what found in the aggregated analysis in Sect. 2; thus, for the sake of completeness, in Fig. 8 the same results of Fig. 7 are shown by gender so that the different ways how Males and Females (in %) allocate the seven sports to the Individual clusters can be appreciated and compared with the results in Sect. 2.

The optimal weights of the Root clusters are $\mathbf{w} = (0.12, -, 0)$, confirming that the baseline similarity between sports is larger in R_1 than in R_3. The weight w_3 is null here indicating that there is no agreement in evaluating the two sports in R_3 as similar, but such a similarity is differently qualified by the subjects (Fig. 9). The optimal weights of the Individual clusters are displayed in Fig. 9 where it is clear that there is a general agreement in evaluating the similarities of the sports within clusters except for students 3 (Individual cluster 2), 6 (Individual cluster 1), 11 (Individual clusters 2 and 3), 12 and 13 (Individual cluster 3). Note that in this application, the proximity matrices are binary: Thus, since the similarities to be fitted are either zeros or ones, in this special case the smaller weights for such students indicate that their similarity matrices are fitted worse than the others.

As for the goodness of fit, the solution analyzed accounts for the 94.56% of the total sum of squares (the relative loss function value is $ESS/TSS = 0.0544$) and can be decomposed into the Root and Individual Within, plus the Between Sum of Squares as follows: $CWSS/TSS = 0.2709$, $IWSS/TSS = 0.6725$, $BSS/TSS = 0.0022$, which confirm how the model optimally fits the data.

For comparison, the INDCLUS model has been also fitted with $J = 3$ and the best solution retained over 200 random starts gives a relative loss equal to 0.3401. The clusters of sports result

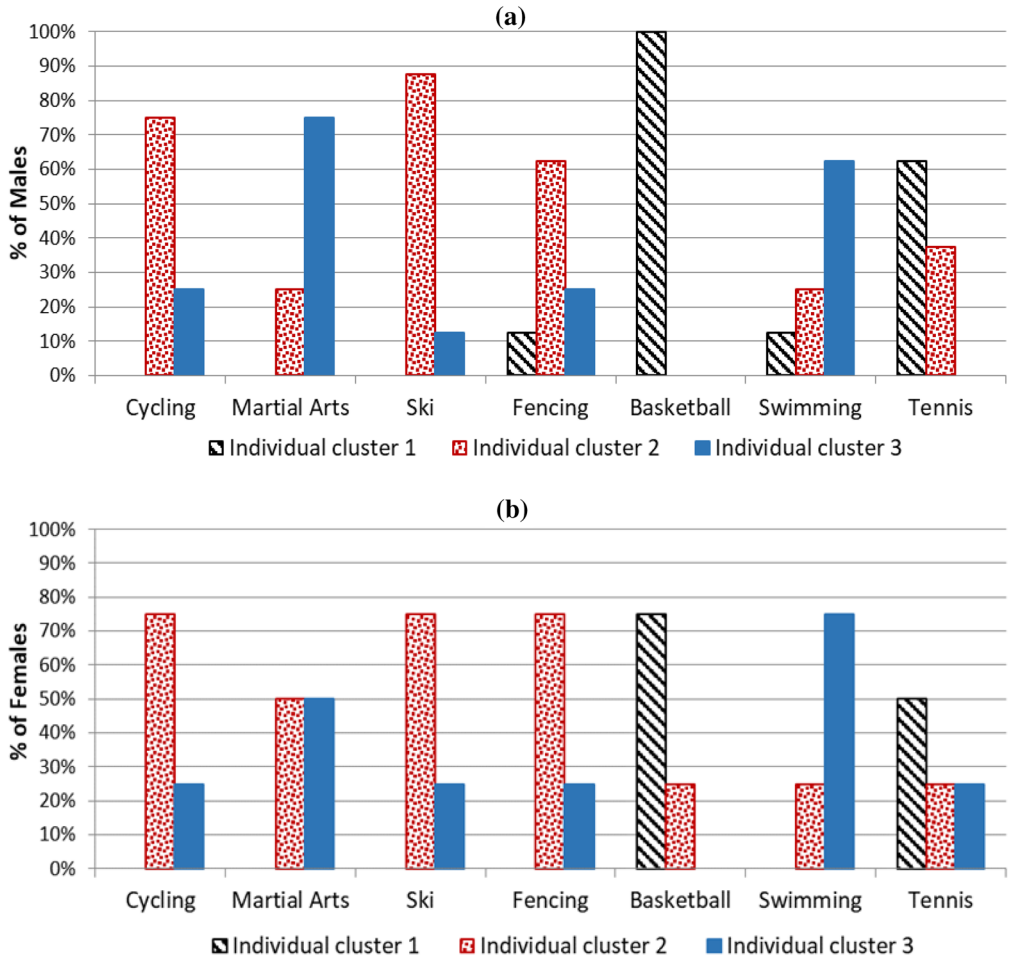

FIGURE 8.
Sport data: sports in individual clusters from ROOTCLUS (**a** % of Males, **b** % of Females).

to be C_1 =(Volleyball, Water Polo, Rugby, Soccer, Basketball), C_2=(Volleyball, Water Polo, Rugby, Soccer, Basketball, *Tennis*), C_3 =(Horse-riding, Cycling, Athletics, Martial Arts, Ski, Fencing, Swimming, Artistic Gym). The flexibility of INDCLUS in allowing for overlapping clusters permits the first two clusters to be identical up to Tennis confirming the double view of a *Team Sport* cluster and a *Ball Sport* cluster, while the third is a residual cluster of *Other Sports*.

From the individual weights in Table 14, two sets of profiles occur where the weights assigned to cluster 1 (Cluster 2) are either almost zero (one) or close to one (zero) but they cannot be explained by gender differences. The weights of cluster 3 are all smaller and probably reflect the heterogeneity of the sports it contains.

7. Discussion

A model for clustering objects in three-way proximity data (termed ROOTCLUS) has been proposed which jointly search for A) *Root* clusters which subsume the similarities on the common

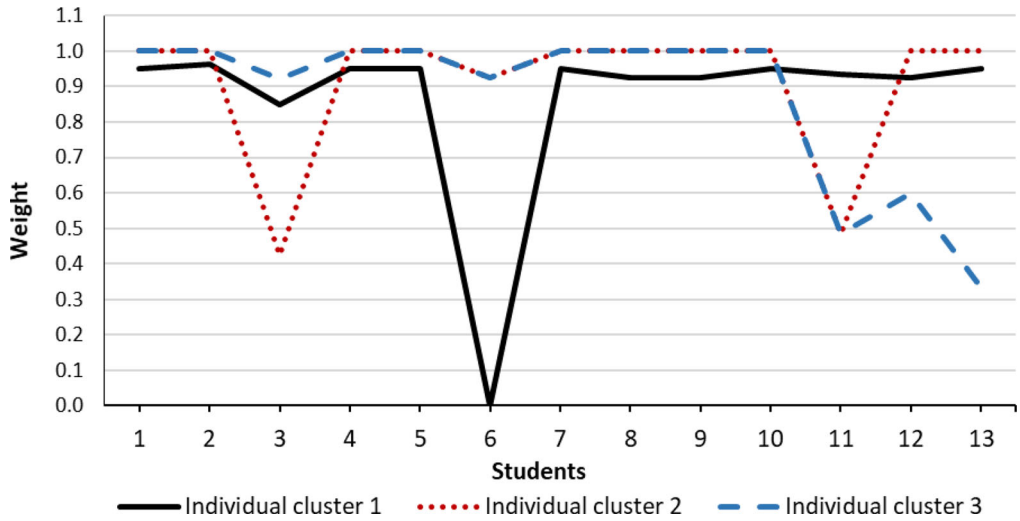

FIGURE 9.
Sports data: weights of the 13 students for each Individual cluster.

TABLE 14.
Sport data: weights resulting from INDCLUS

C_1	C_2	C_3	Gender
0.83	0.00	0.15	M
0.00	0.87	0.20	M
0.96	0.00	0.53	M
0.00	1.00	0.43	F
0.00	1.00	0.43	M
0.45	0.00	0.31	F
0.00	1.00	0.43	F
0.96	0.00	0.53	F
0.92	0.00	0.35	M
0.00	1.00	0.25	M
0.94	0.00	0.26	F
0.00	1.00	0.29	M

perception of the subjects about a subset of objects; B) *Individual* clusters of the remaining objects accounting for the heterogeneity of the subjects in evaluating the similarities.

The model is flexible and allows to account for the subject perception in evaluating and clustering the similarity between objects. As evident from the applications on real data, the complementary (Root and Individual) partitions define corresponding clusters with subject-specific meaning which reflect different profiles, provide an interpretively satisfying approach in several contexts where the subjects' perceptions are differently expressed and allows to identify subsets of objects that are exclusive to certain subjects or categories of subjects.

Moreover, the results of the extensive simulation study demonstrate the effectiveness of the proposed method and its performance in different conditions.

As for the algorithm, different choices of the random starts and further analyses of its efficiency and capability to recover the correct partitions deserve more investigations in different and more

complex situations. Moreover, the performance when different number of clusters J are chosen also deserves to be analyzed.

As for the model assumptions, the overlap of the clusters could be allowed (as in INDCLUS) to guarantee a major flexibility at the cost of an increase in model complexity. Conversely, a more parsimonious model may be derived by considering classes of subjects sharing the same "Individual" partitions to identify similar profiles of subjects.

There is still room for extensions regarding the use of possible supplementary information to better interpret and describe the Individual clusters of objects even for the prediction of the perceptual profile of additional subjects.

Acknowledgments

The authors are grateful to the Associate Editor and referees for their valuable comments and suggestions which greatly improved the presentation and content of the first version.

Appendix

In order to solve the constrained problem (14) in those cases when the diagonal entries of the H similarity matrices $\mathbf{S}_h$ are not of interest, the steps 1 to 4 of the ALS-type algorithm presented in Sect. 4 can be modified straightforwardly as follows.

Since only the off-diagonal elements of matrices $\mathbf{S}_h$ $(h = 1, \ldots, H)$ need to be considered, the loss function (14) becomes

$$f_{\text{off}}(\mathbf{P}, \mathbf{W}, \mathbf{M}_h, \mathbf{V}_h, c_h) = \; F_{\text{off}}(\mathbf{P}, \mathbf{W}, \mathbf{M}_h, \mathbf{V}_h, c_h) + \lambda G, \tag{23}$$

where

$$F_{\text{off}}(\mathbf{P}, \mathbf{W}, \mathbf{M}_h, \mathbf{V}_h, c_h) = \frac{\sum_{h=1}^{H} \left\| \mathbf{S}_h - \mathbf{PWP}' - \left(\mathbf{P} + \mathbf{M}_h\right)\mathbf{V}_h\left(\mathbf{P} + \mathbf{M}_h\right)' - c_h \mathbf{1}_N \mathbf{1}_N' \right\|_{\text{off}}^2}{\sum_{h=1}^{H} \|\mathbf{S}_h\|_{\text{off}}^2} \tag{24}$$

and $\|\mathbf{Z}\|_{\text{off}}^2 = \sum_{x=1}^{X} \sum_{y=1\,;\, y\neq x}^{Y} z_{xy}^2$.

In step 1, the loss function (23), instead of (14), is minimized over $\mathbf{P}$ and $\mathbf{M}_h$ $(h = 1, \ldots, H)$.

In steps 2 to 4, all the rows of $\mathbf{s}_h$, $\mathbf{T}$, $\mathbf{Q}_h$ and $\mathbf{1}_{N^2}$ in model (15), corresponding to the diagonal entries of the matrices in (14), need to be left out. Such reduced structures are obtained as follows:

$$\tilde{\mathbf{s}}_h = \mathbf{s}_h \odot \mathbf{d} \qquad (h = 1, \ldots, H)\,, \tag{25}$$

$$\tilde{\mathbf{T}} = \mathbf{T} \odot \mathbf{D}\,, \tag{26}$$

$$\tilde{\mathbf{Q}}_h = \mathbf{Q}_h \odot \mathbf{D} \qquad (h = 1, \ldots, H)\,, \tag{27}$$

$$\tilde{\mathbf{1}}_{N^2} = \mathbf{1}_{N^2} \odot \mathbf{d}\,, \tag{28}$$

where $\odot$ denotes the Hadamard product, $\mathbf{d}$ is the column vector of size N^2 of the vectorized matrix $\left(\mathbf{1}_N \mathbf{1}_N' - \mathbf{I}_N\right)$, being $\mathbf{I}_N$ the identity matrix of size N, and $\mathbf{D}$ is the $N^2 \times J$ matrix having all its columns equal to $\mathbf{d}$.

Therefore, model (15) is rewritten in terms of (25)–(28) and Steps 2, 3 and 4 modified accordingly.

References

Bocci, L., & Vicari, D. (2017). GINDCLUS: Generalized INDCLUS with external information. *Psychometrika*, *82*, 355–381.

Bocci, L., Vicari, D., & Vichi, M. (2006). A mixture model for the classification of three-way proximity data. *Computational Statistics & Data Analysis*, *50*, 1625–1654.

Calinski, T., & Harabasz, J. (1974). A dendrite method for cluster analysis. *Communications in Statistics*, *3*, 1–27.

Carroll, J. D., & Arabie, P. (1983). INDCLUS: An individual differences generalization of ADCLUS model and the MAPCLUS algorithm. *Psychometrika*, *48*, 157–169.

Carroll, J. D., & Chang, J. J. (1970). Analysis of individual differences in multidimensional scaling via an N-generalization of the Eckart–Young decomposition. *Psychometrika*, *35*, 283–319.

Chaturvedi, A., & Carroll, J. D. (2006). CLUSCALE (CLUstering and multidimensional SCAL[E]ing): A three-way hybrid model incorporating clustering and multidimensional scaling structure. *Journal of Classification*, *23*, 269–299.

Chaturvedi, A. J., & Carroll, J. D. (1994). An alternating combinatorial optimization approach to fitting the INDCLUS and generalized INDCLUS models. *Journal of Classification*, *11*, 155–170.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, *20*, 37–46.

De Leeuw, J. (1994). Block-relaxation algorithms in statistics. In H. H. Bock, W. Lenski, & M. M. Richter (Eds.), *Information systems and data analysis* (pp. 308–325). Berlin: Springer.

Giordani, P., & Kiers, H. A. L. (2012). FINDCLUS: Fuzzy INdividual Differences CLUStering. *Journal of Classification*, *29*, 170–198.

Gordon, A. D., & Vichi, M. (1998). Partitions of Partitions. *Journal of Classification*, *15*, 265–285.

Hubert, L. J., & Arabie, P. (1985). Comparing partitions. *Journal of Classification*, *2*, 193–218.

Hubert, L. J., Arabie, P., & Meulman, J. (2006). *The structural representation of proximity matrices with MATLAB*. Philadelphia: SIAM.

Kiers, H. A. L. (1997). A modification of the SINDCLUS algorithm for fitting the ADCLUS and INDCLUS models. *Journal of Classification*, *14*, 297–310.

Lawson, C. L., & Hanson, R. J. (1974). *Solving least squares problems*. Englewood Cliffs: Prentice Hall.

McDonald, R. P. (1980). A simple comprehensive model for the analysis of covariance structures: Some remarks on applications. *British Journal of Mathematical and Statistical Psychology*, *33*, 161–183.

Mirkin, B. G. (1987). Additive clustering and qualitative factor analysis methods for similarity matrices. *Journal of Classification*, *4*, 7–31.

Rao, C. R., & Mitra, S. (1971). *Generalized inverse of matrices and its applications*. New York: Wiley.

Rocci, R., & Vichi, M. (2008). Two-mode multi-partitioning. *Computational Statistics & Data Analysis*, *52*, 1984–2003.

Shepard, R. N., & Arabie, P. (1979). Additive clustering: Representation of similarities as combinations of discrete overlapping properties. *Psychological Review*, *86*, 87–123.

Schepers, J., Ceulemans, E., & Van Mechelen, I. (2008). Selecting among multi-mode partitioning models of different complexities: A comparison of four model selection criteria. *Journal of Classification*, *25*, 67–85.

Schiffman, S. S., Reynolds, M. L., & Young, F. W. (1981). *Introduction to multidimensional scaling*. London: Academic Press.

Vicari, D., & Vichi, M. (2009). Structural classification analysis of three-way dissimilarity data. *Journal of Classification*, *26*, 121–154.

Vichi, M. (1999). One mode classification of a three-way data set. *Journal of Classification*, *16*, 27–44.

Wedel, M., & DeSarbo, W. S. (1998). Mixtures of (constrained) ultrametric trees. *Psychometrika*, *63*, 419–443.

Wilderjans, T. F., Depril, D., & Van Mechelen, I. (2012). Block-relaxation approaches for fitting the INDCLUS model. *Journal of Classification*, *29*, 277–296.

Manuscript Received: 4 JUL 2018
Final Version Received: 13 AUG 2019
Published Online Date: 13 SEP 2019