



# A between-cluster approach for clustering skew-symmetric data

Donatella Vicari<sup>1</sup> · Cinzia Di Nuzzo<sup>1,2</sup>

Received: 2 January 2023 / Accepted: 1 October 2023  
© The Author(s) 2023

## Abstract

In order to investigate exchanges between objects, a clustering model for skew-symmetric data is proposed, which relies on the between-cluster effects of the skew-symmetries that represent the imbalances of the observed exchanges between pairs of objects. The aim is to detect clusters of objects that share the same behaviour of exchange so that origin and destination clusters are identified. The proposed model is based on the decomposition of the skew-symmetric matrix pertaining to the imbalances *between* clusters into a sum of a number of off-diagonal block matrices. Each matrix can be approximated by a skew-symmetric matrix by using a truncated Singular Value Decomposition (SVD) which exploits the properties of the skew-symmetric matrices. The model is fitted in a least-squares framework and an efficient Alternating Least Squares algorithm is provided. Finally, in order to show the potentiality of the model and the features of the resulting clusters, an extensive simulation study and an illustrative application to real data are presented.

**Keywords** Clustering · Skew-symmetric data · SVD · Asymmetry · Dimension reduction

**Mathematics Subject Classification** 62Hxx Multivariate analysis

---

✉ Donatella Vicari  
donatella.vicari@uniroma1.it

Cinzia Di Nuzzo  
cinzia.dinuzzo@unict.it

<sup>1</sup> Department of Statistical Sciences, Sapienza University of Rome, Pl. A. Moro, 5, Roma 00185, Italy

<sup>2</sup> Department of Economics and Business, University of Catania, Corso Italia, 55, Catania 95129, Italy

## 1 Introduction

Asymmetric data pertaining to pairwise exchanges or flows between objects are observed and analysed in order to investigate on their intrinsic asymmetry; some examples are commercial exchanges, brand switching, migration data, or confusion data. Several methodologies have been proposed to deal with such an asymmetry when the main interest is to investigate on the directions of the exchanges. Specifically, relying on the decomposition of the asymmetry into symmetric and skew-symmetric effects, many models have been proposed which either estimate only the skew-symmetric part or fit the two components.

In order to visualise and explore asymmetric data, different methodologies have been introduced; in particular, in Gower (1977) and Constantine and Gower (1978) a decomposition of any asymmetric matrix is introduced which allows to obtain a graphical representation of the objects on a plane where the areas of the triangles formed by all triplets of objects are proportional to the amounts of the imbalances of the exchanges observed between objects. In addition, to jointly display the symmetric and skew-symmetric components of the data, several models have been proposed, e.g. Zielman and Heiser (1996); Rocci and Bove (2002); Bove and Okada (2018); Bove and Vicari (2023) and, for an extensive review, see also Saito and Yadohisa (2005) and Bove et al. (2021).

In a non-hierarchical clustering context, an asymmetric version of the  $k$ -means algorithm is proposed in Olszewski (2012), while a centroid-based approach is proposed in Olszewski and Ster (2014) using an asymmetric dissimilarity.

In this work, the focus concerns the analysis of the imbalances of the exchanges observed between pairs of objects that are described by the skew-symmetric component of the data and the main interest is to investigate the exchanges or flows between data. A skew-symmetric matrix can derive from the skew-symmetric component of an observed asymmetric proximity matrix (Gower 1977), or from a transformation of the observed asymmetries that incorporates the average amounts of the data (Saito and Yadohisa 2005). In some cases skew-symmetric data can be observed directly, such as where the symmetric component of an observed asymmetric matrix is constant and therefore noninformative. Again, when the data itself is skew-symmetric such as, for example, in the analysis of comparative judgments in which an individual could be asked to evaluate the difference between pairs of stimuli and to make a judgment on their degree of preference.

Note that "imbalances" play a fundamental role in this context, because the departures from symmetry, in terms of magnitudes and signs, provide information on both the intensities and directions of the exchanges between objects. For this reason, analysing imbalances is the main objective of clustering asymmetric data.

Cluster analysis methodologies for skew-symmetric data search for either dominant objects or clusters of objects with similar behaviours in terms of both magnitude and direction of the exchanges. The basic idea is that clusters of objects can share common behaviours not only in terms of average intensities but also in the directions of the exchanges: some clusters of objects can mainly either originate exchanges directed to other clusters or receive from other clusters.

In order to account for the within- and between- cluster effects, a different approach has been proposed in Vicari (2014) where a partition of the objects is jointly identified from the symmetric and skew-symmetric components of the data. In Vicari (2018) an extension of model (Vicari 2014) has been presented by introducing the possibility to incorporate external variables in order to explain the imbalances between objects. Within the same framework, in Vicari (2020) an alternative model is proposed which jointly fits both the symmetric and the skew-symmetric component by using two clustering structures depending on two partitions of objects: a complete (standard) partition, from the symmetric component, and an incomplete partition, to fit the skew-symmetric component, which is nested into the complete one and where objects are allowed to remain possibly unassigned.

In this paper, a new clustering model for skew-symmetric data is proposed, which aims to detect clusters of objects that share the same behaviour of exchange in terms of both amounts and directions so that *origin* and *destination* clusters can be identified.

The model relies on the decomposition of the skew-symmetric matrix into a sum of a number of off-diagonal block matrices which contain the pairwise exchanges between clusters. They are optimally reconstructed in the least-squares sense by using separate truncated Singular Value Decompositions (SVD) which provide their best low-rank matrix approximations.

Interestingly, since the resulting singular vectors allow objects to be mapped into low-dimensional (possibly two-dimensional) spaces, graphical representations can facilitate the interpretation of the exchanges between objects.

The model is fitted in a least-squares framework and an efficient Alternating Least Squares (ALS) algorithm is provided.

The rest of the paper is organized as follows. In order to motivate our model, an illustrative example is described in Sect. 2. The clustering model is introduced and formalized in Sect. 3 and, in order to fit the model, an appropriate ALS algorithm is provided in Sect. 4. In Sect. 5 an extensive simulation study is carried out on artificial data to assess the potentiality and the effectiveness of the proposal. In Sect. 6 an application to real data is presented. Finally in Sect. 7 some concluding remarks are provided.

## 2 Illustrative example

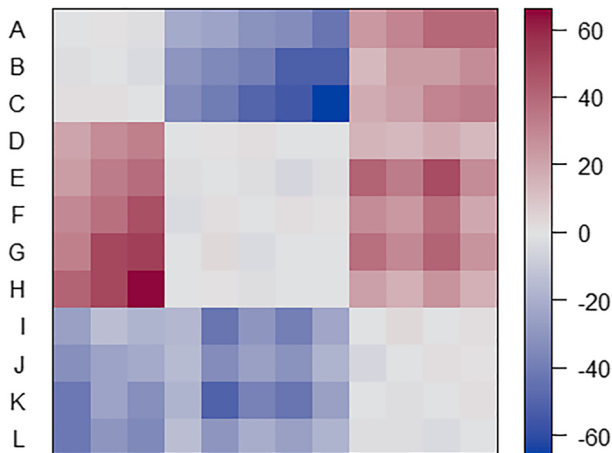
In order to give both a flavour of the problem dealt with and an intuition of the model fully formalized in Sect. 3, we consider here an artificial example with 12 objects (denoted by *A-L*) which can motivate the method and illustrate its features and potentiality.

Without loss of generality, let us suppose that the artificial ( $N \times N$ ) skew-symmetric matrix  $\mathbf{K} = (k_{ij})_{i, j=1, \dots, N}$  in Table 1 contains the imbalances observed or derived from asymmetric dissimilarity data pertaining to pairwise exchanges between 12 objects. From the heat map in Fig. 1 associated with the data in Table 1, some features of the objects are evident:

**Table 1** Artificial skew-symmetric data

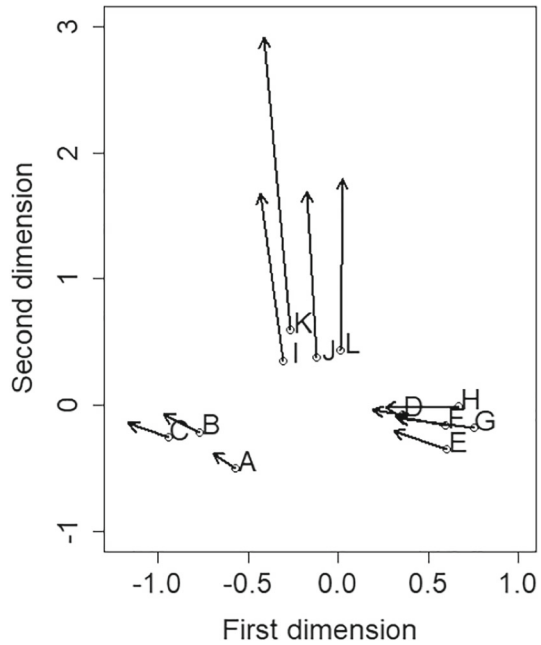
<b>K</b>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>	<i>I</i>	<i>J</i>	<i>K</i>	<i>L</i>
<i>A</i>	0	-1	2	21	24	30	33	43	-25	-32	-41	-41
<i>B</i>	1	0	3	29	35	39	52	52	-14	-24	-24	-29
<i>C</i>	-2	-3	0	33	40	49	55	66	-18	-22	-32	-35
<i>D</i>	-21	-29	-33	0	-1	-3	0	0	-16	-15	-18	-14
<i>E</i>	-24	-35	-40	1	0	2	4	1	-43	-34	-51	-29
<i>F</i>	-30	-39	-49	3	-2	0	-3	-1	-29	-25	-38	-20
<i>G</i>	-33	-52	-55	0	-4	3	0	0	-39	-30	-43	-26
<i>H</i>	-43	-52	-66	0	-1	1	0	0	-23	-17	-26	-17
<i>I</i>	25	14	18	16	43	29	39	23	0	-4	0	-2
<i>J</i>	32	24	22	15	34	25	30	17	4	0	-2	-1
<i>K</i>	41	24	32	18	51	38	43	26	0	2	0	-3
<i>L</i>	41	29	35	14	29	20	26	17	2	1	3	0

Skew-symmetric matrix **K** of imbalances between 12 objects

**Fig. 1** Artificial skew-symmetric data. Heatmap of **K**

- $\{A, B, C\}$  have large imbalances both outgoing towards  $\{I, J, K, L\}$  and incoming from  $\{D, E, F, G, H\}$ ;
- $\{I, J, K, L\}$  have only incoming imbalances, i.e., they are destinations from all other objects;
- $\{D, E, F, G, H\}$  have only outgoing imbalances, i.e., they are origins for all other objects;
- only small pairwise imbalances are present within blocks  $\{A, B, C\}$ ,  $\{D, E, F, G, H\}$  and  $\{I, J, K, L\}$ .

**Fig. 2** Artificial skew-symmetric data. Map from metric MDS on  $abs(\mathbf{K})$ . Drift vectors attached to objects are estimated from  $\mathbf{K}$



As a preliminary analysis, Fig. 2 shows the map obtained from the standard metric MultiDimensional Scaling (MDS) of the symmetric matrix  $abs(\mathbf{K})$ , where  $abs$  denotes the absolute value of matrix  $\mathbf{K}$ . In addition to the amounts of the imbalances, information on their directions is incorporated into the map by drawing drift vectors from any object-point having lengths and directions proportional to the average row totals (average outgoing imbalances), see Bove et al. (2021). Thus, the positions of the objects in the map identify three main clusters  $G_1 = \{A, B, C\}$ ,  $G_2 = \{D, E, F, G, H\}$ ,  $G_3 = \{I, J, K, L\}$  having similar amounts of exchange on average. Moreover, in Fig. 2 the directions of the imbalances can also be identified on average: drift vectors of the objects in  $G_2$  point to  $G_1$  and  $G_3$ , while drift vectors in  $G_3$  are directed upwards, because they have incoming imbalances from all other objects.

In summary, from this preliminary analysis it is evident that the artificial data define a situation with a clear clustering structure where objects in the same cluster share common behaviours towards objects in different clusters in terms of amount and direction of the imbalances, i.e., objects exhibit large imbalances directed towards objects in different clusters (*between imbalances*) and small imbalances with objects in the same cluster (*within imbalances*).

In order to identify such a clustering structure of the data, the model proposed in Vicari (2018) has been applied to the skew-symmetric matrix  $\mathbf{K}$  in Table 1 and the best three-cluster solution (goodness-of-fit<sup>1</sup> equal to 93%), retained in

<sup>1</sup> goodness-of-fit =  $\left(1 - \frac{\|\mathbf{K} - \widehat{\mathbf{K}}\|^2}{\|\mathbf{K}\|^2}\right) 100$ , where hereafter  $\widehat{\mathbf{K}}$  generally denotes the skew-symmetric matrix estimated by the model.

100 runs of the algorithm from different random starts, gives the partition  $G_1^* = \{A, B, C, D, E, F, G, H\}$ ,  $G_2^* = \{I, J, L\}$  and  $G_3^* = \{K\}$ .

Model in Vicari (2018) returns vector  $\mathbf{b} = (-9.27, -9.27, -9.27, -9.27, -9.27, -9.27, -9.27, 17.17, 17.17, 22.67, 17.17)$  with as many different values as there are clusters representing the average imbalances of all objects of a cluster towards any other object not belonging to the same cluster. Therefore, vector  $\mathbf{b}$  reveals the directions:  $G_1^*$  results to be an origin cluster, while  $G_2^*$  and  $G_3^*$  are two destination clusters. Therefore, this model fails to fully capture the underlying clustering structure, as it is unable to reveal the differences between  $\{A, B, C\}$  and  $\{D, E, F, G, H\}$  in terms of directions of their imbalances.

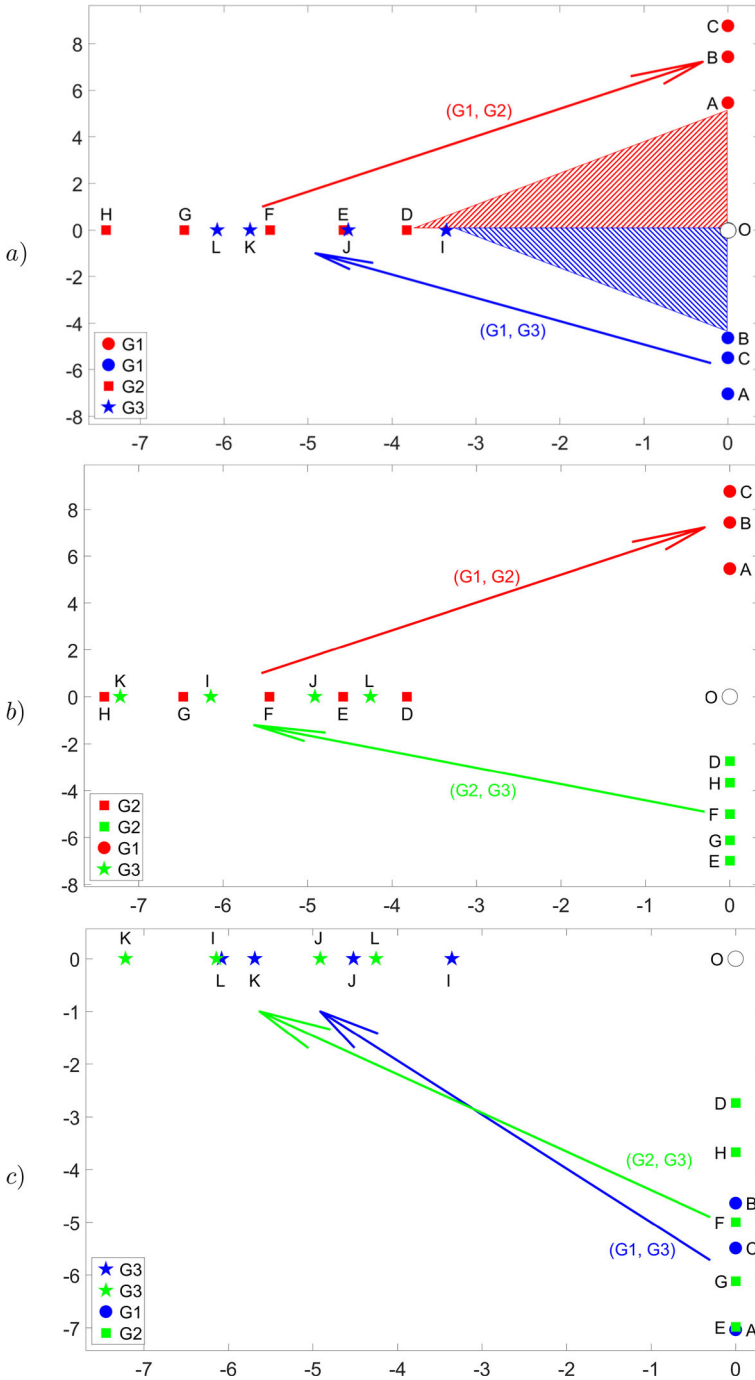
In order to detect the underlying clustering structure and better explain the between-cluster variability, the model proposed here, which will be fully formalized in Sect. 3, has been fitted to the artificial data of Table 1. The best resulting partition (obtained in different runs of the algorithm from 100 random starts) correctly identifies clusters  $G_1$ ,  $G_2$  and  $G_3$  with a goodness-of-fit equal to 99.6%. Interestingly, given the optimal partition into  $C$  clusters, the proposed model provides as output pairs of singular vectors that allow to graphically represent the objects in  $C$  Gower diagrams (Gower 2018), from which useful information is obtained about the amount and direction of the skew-symmetries between clusters. Specifically, for any cluster  $G_c$  it is possible to map all objects belonging to either  $G_c$  or any other cluster  $G_{\tilde{c}}$  ( $c, \tilde{c} = 1, \dots, C$ ,  $c < \tilde{c}$ ) onto a low-dimensional space (a plane in this case) spanned by the optimal pairs of singular vectors where the directions of the exchanges between objects in  $G_c$  and  $G_{\tilde{c}}$  can be identified.

In general, one major advantage of the Gower diagrams (Gower 2018) is the interpretation in terms of the area of the triangle with vertices corresponding to any pair of objects and the origin  $O$  (see Fig. 3a), because such area is approximatively proportional to the size of their pairwise skew-symmetry.

As an example from the proposed model, Fig. 3a displays the map of the pairwise imbalances (1) between objects in cluster  $G_1$  (red circles) and objects in clusters  $G_2$  (red squares) and (2) between objects in cluster  $G_1$  (blue circles) and objects in cluster  $G_3$  (blue stars). Therefore, for example, the area of the red triangle ( $O, A, D$ ) is proportional to the size of the imbalance  $k_{AD}$  (Table 1).

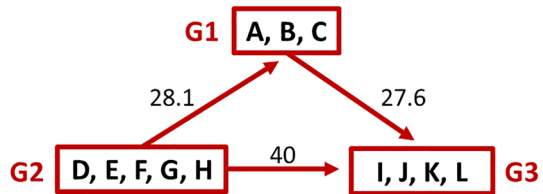
We may observe that two close points represent small values of skew-symmetry because the area of the corresponding triangle is small, as well as pairs of points either nearly collinear with or very close to the origin also display small skew-symmetries. Conversely, pairs of points far from the origin represent large imbalances (for further details, see Gower 1977, 2018).

Moreover, since triangles ( $O, A, D$ ) and ( $O, D, A$ ) are the same but with their labels permuted and give the same areas, by convention, a clockwise direction denotes a negative area, while an anticlockwise direction indicates a positive area (in accordance with the basic property of skew-symmetric matrices that  $k_{AD} = -k_{DA}$ ). Thus, for example, the red triangle ( $O, A, D$ ) has an area greater than the blue triangle ( $O, B, I$ ), i.e., regardless of the sign, the imbalance between  $A$  and  $D$  is greater than that between  $B$  and  $I$ ; as for the directionality,  $k_{DA}$  is negative ( $D$  together with all objects in  $G_2$  is the origin for all objects in  $G_1$ ), while  $k_{IB}$  is positive ( $I$  is the destination for  $G_1$  as well as all objects in  $G_3$ ).



**Fig. 3** Artificial skew-symmetric data. Scatter plot of objects for cluster: **a**  $G_1$ ; **b**  $G_2$ ; **c**  $G_3$ . Directions of imbalances between clusters are represented by arrows. Areas of triangles represent skew-symmetries between objects

**Fig. 4** Artificial skew-symmetric data. Summary graph of the clusters with average fitted imbalances between clusters



The three Gower diagrams for the optimal three-cluster solution are displayed in Fig. 3. The directions of the exchanges between cluster  $G_1$  and the others are shown in Fig. 3a: cluster  $G_1 = \{A, B, C\}$  (blue circles) has negative skew-symmetries with cluster  $G_3 = \{I, J, K, L\}$  (blue stars), i.e.  $\{A, B, C\}$  have outgoing flows towards  $\{I, J, K, L\}$ , while  $G_1 = \{A, B, C\}$  (red circles) has positive skew-symmetries with  $G_2 = \{D, E, F, G, H\}$  (red squares), i.e.  $\{A, B, C\}$  have incoming flows from  $\{D, E, F, G, H\}$ . Similarly, the amounts and directions of the imbalances between either cluster  $G_2$  or  $G_3$  and the others can be derived from Figs. 3b and c, respectively.

Finally, the summary graph in Fig. 4 reports the average fitted imbalances between clusters to highlight how the proposed model is able to account for the relationships between objects in terms of between-cluster exchanges by correctly reconstructing the imbalances in Table 1.

Note that since the model presented in Sect. 3 accounts for the between-cluster variability, the resulting Gower diagrams in Fig. 3 have some nice peculiar features:

- objects within the same cluster have the same coordinate (either abscissa or ordinate) equal to zero, i.e., they all lie either on the  $x$ - or  $y$ -axis;
- all pairs of objects in different clusters generate right-angled triangles;
- objects within the same cluster determine degenerate triangles with null areas;
- all objects in a cluster generate imbalances of the same sign towards objects in a different cluster, i.e., all objects within a cluster share a common exchange behaviour towards objects in another cluster so that origin and destination clusters can be identified.

To sum up, the model, which is fully formalised and discussed in the next Sections, provides a partition of the objects and a reduced number of dimensions for each cluster (two dimensions in this application) which allow to map the objects and represent the imbalances between all objects in one cluster and all objects in different clusters in terms of right-angled triangles areas.

**Remark 1** It is worth noting that when the observed asymmetric data are similarities (instead of dissimilarities), the signs of the imbalances take on the reverse meaning. Therefore, in Gower diagrams derived from similarity data a clockwise direction from one object to another denotes a positive area, i.e., a positive imbalance, which qualifies the first object as the origin.

### 3 The model

In this Section the between-cluster model for skew-symmetric data is formalised.



Let  $\Omega = \{\omega_{ij}\}$  be an  $(N \times N)$  asymmetric matrix, which can be uniquely decomposed into a sum of a symmetric matrix  $\mathbf{S}$  and a skew-symmetric matrix  $\mathbf{K}$  as follows

$$\Omega = \mathbf{S} + \mathbf{K} = \frac{1}{2}(\Omega + \Omega^T) + \frac{1}{2}(\Omega - \Omega^T),$$

where  $\mathbf{S}$  and  $\mathbf{K}$  have size  $(N \times N)$  and are orthogonal to each other, i.e.,  $tr(\mathbf{SK}) = 0$ . The entry  $s_{ij} \in \mathbf{S}$  represents the average amount of the exchange between objects  $i$  and  $j$ , while the entry  $k_{ij} \in \mathbf{K}$  represents the imbalance between  $i$  and  $j$ , i.e., the amount by which  $k_{ij}$  differs from the mean  $s_{ij}$  ( $i, j = 1, \dots, N$ ). Thus the skew-symmetric matrix  $\mathbf{K} = (k_{ij})$  is such that  $k_{ij} = -k_{ji}$  by definition.

The goal is to cluster the  $(N \times N)$  skew-symmetric matrix  $\mathbf{K}$  by considering a partition of the  $N$  objects into  $C$  disjoint clusters which can be identified by an  $(N \times C)$  binary membership matrix  $\mathbf{U} = (u_{ic})$  such that  $u_{ic} = 1$  if  $i$  belongs to cluster  $c$ ,  $u_{ic} = 0$  otherwise, for  $c = 1, \dots, C$  and  $\sum_{c=1}^C u_{ic} = 1$  for all  $i = 1, \dots, N$ .

Note that, given a partition  $\mathbf{U}$ , matrix  $\mathbf{K}$  can be decomposed as the sum of its *within* and *between* parts, as follows

$$\mathbf{K} = \mathbf{B} + \mathbf{W}, \tag{1}$$

where  $\mathbf{B}$  is an  $(N \times N)$  skew-symmetric off-diagonal block matrix depending on partition  $\mathbf{U}$  and represents the imbalances *between* clusters;  $\mathbf{W}$  is the  $(N \times N)$  skew-symmetric block diagonal matrix of the imbalances *within* clusters. As an example, for the sake of clarity, Fig. 5 graphically shows the decomposition (1) of a matrix  $\mathbf{K}$  into its within and between components for a given partition into three clusters.

Here, we are interested in modelling the between part of the exchanges. The idea is to identify clusters of objects having similar behaviour in terms of amounts and directions of the imbalances directed towards other clusters so that each cluster is mainly either origin or destination.

In order to identify the directed relationships *between* clusters which can be possible origins/destinations, the between component  $\mathbf{B}$  is modelled as follows.

Let  $\mathcal{C} = \{1, \dots, c, \dots, C\} \subset \mathbb{N}$  be the set of the indices of the clusters and  $\mathcal{G} = \{G_1, \dots, G_c, \dots, G_C\}$  be the set of the clusters.

We can consider the  $(N \times N)$  skew-symmetric matrix  $\mathbf{B}^{(c, \tilde{c})}$  ( $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) of the imbalances between any pair of clusters  $G_c$  and  $G_{\tilde{c}}$  which has all elements equal to zero except for two rectangular blocks corresponding to the objects belonging to cluster either  $G_c$  or  $G_{\tilde{c}}$  (see Fig. 5 as an example). Note that all matrices  $\mathbf{B}^{(c, \tilde{c})}$  ( $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) are orthogonal to each other by construction.

Let us recall that, due to its special form, any skew-symmetric matrix of size  $N$  can always be decomposed in canonical form as the sum of a number of skew-symmetric matrices of rank 2 (Gower 2018) by using its  $\lfloor \frac{N}{2} \rfloor$  distinct singular values  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{\lfloor N/2 \rfloor}$ , where  $\lfloor \cdot \rfloor$  denotes the integer part (see Appendix 1 for details). Specifically, in the view of a dimension reduction and given a partition  $\mathbf{U}$ , any matrix

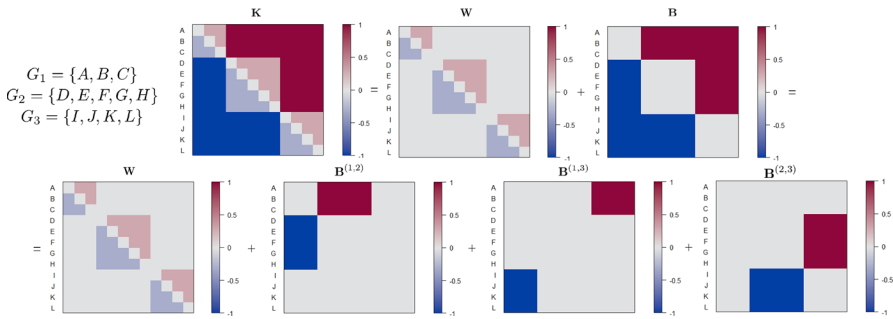


Fig. 5 Decomposition of matrix  $\mathbf{K}$  for a given partition into three clusters

$\mathbf{B}^{(c, \tilde{c})}$  can be optimally approximated by using the truncated SVD:

$$\mathbf{B}^{(c, \tilde{c})} = \mathbf{P}_{(R)}^{(c, \tilde{c})} \mathbf{\Lambda}_{(R)}^{(c, \tilde{c})} \mathbf{J} \mathbf{P}_{(R)}^{(c, \tilde{c})\top} + \mathbf{\Xi}_{(R)}^{(c, \tilde{c})}, \quad c, \tilde{c} \in \mathcal{C}, c < \tilde{c}, \quad R \leq \left\lfloor \frac{N}{2} \right\rfloor, \quad (2)$$

where  $\mathbf{P}_{(R)}^{(c, \tilde{c})}$  denotes the  $(N \times 2R)$  matrix of the first  $2R$  left singular vectors of  $\mathbf{B}^{(c, \tilde{c})}$ ,  $\mathbf{\Lambda}_{(R)}^{(c, \tilde{c})}$  is the  $(2R \times 2R)$  diagonal matrix with elements equal to the first  $2R$  singular values of  $\mathbf{B}^{(c, \tilde{c})}$ ,  $\mathbf{J}$  is an  $(N \times N)$  block diagonal matrix with matrices  $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$  along its diagonal and  $\mathbf{\Xi}_{(R)}^{(c, \tilde{c})}$  is the  $(N \times N)$  residual matrix of the truncated SVD.

Note that the theoretical justification of the use of the SVD in (2) comes from the well-known general result of Eckart and Young (1936) on the problem of matrix approximation of reduced rank, whose least-squares solution turns out to be the SVD.

Thus, the off-diagonal matrix  $\mathbf{B}$  of the imbalances between clusters can be expressed as the sum of all approximated matrices  $\mathbf{B}^{(c, \tilde{c})}$  ( $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) for all pairs of clusters, i.e.,

$$\mathbf{B} = \sum_{c, \tilde{c} \in \mathcal{C}, c < \tilde{c}} \mathbf{B}^{(c, \tilde{c})} = \quad (3)$$

$$= \sum_{c, \tilde{c} \in \mathcal{C}, c < \tilde{c}} \sum_{n=1}^R \lambda_n^{(c, \tilde{c})} \left( \mathbf{p}_{2n-1}^{(c, \tilde{c})} \mathbf{p}_{2n}^{(c, \tilde{c})\top} - \mathbf{p}_{2n}^{(c, \tilde{c})} \mathbf{p}_{2n-1}^{(c, \tilde{c})\top} \right) + \mathbf{\Xi} = \quad (4)$$

$$= \sum_{c, \tilde{c} \in \mathcal{C}, c < \tilde{c}} \sum_{n=1}^R \left( \mathbf{v}_{2n-1}^{(c, \tilde{c})} \mathbf{v}_{2n}^{(c, \tilde{c})\top} - \mathbf{v}_{2n}^{(c, \tilde{c})} \mathbf{v}_{2n-1}^{(c, \tilde{c})\top} \right) + \mathbf{\Xi}, \quad (5)$$

where  $\mathbf{\Xi}$  is the  $(N \times N)$  residual matrix due to the truncated SVD,  $\lambda_n^{(c, \tilde{c})}$  ( $n = 1, \dots, R$ ) is the  $n$ -th distinct singular value of  $\mathbf{B}^{(c, \tilde{c})}$ ,  $\mathbf{p}_j^{(c, \tilde{c})}$  ( $j = 1, \dots, 2R$ ) is the  $j$ -th left

singular vector of  $\mathbf{B}^{(c, \tilde{c})}$ , and

$$\mathbf{v}_{2n-1}^{(c, \tilde{c})} = \sqrt{\lambda_n^{(c, \tilde{c})}} \mathbf{p}_{2n-1}^{(c, \tilde{c})}, \quad \mathbf{v}_{2n}^{(c, \tilde{c})} = \sqrt{\lambda_n^{(c, \tilde{c})}} \mathbf{p}_{2n}^{(c, \tilde{c})}, \quad (n = 1, \dots, R). \quad (6)$$

It is worth noting that  $\mathbf{p}_{2n-1}^{(c, \tilde{c})}$  and  $\mathbf{p}_{2n}^{(c, \tilde{c})}$  are the singular vectors associated to the  $n$ -th singular value  $\lambda_n^{(c, \tilde{c})}$  ( $n = 1, \dots, R$ ) of  $\mathbf{B}^{(c, \tilde{c})}$ , see Appendix 1 for details.

As an approximation of  $\mathbf{B}$ , model (5) optimally reconstructs, in the least-squares sense, the imbalances between objects belonging to different clusters in a low-dimensional space by using  $R$  *bimensions* (see Appendix 1).

Note that we may also consider the full SVD of any skew-symmetric matrix  $\mathbf{B}^{(c, \tilde{c})}$  which derives from (5) by setting  $R = \lfloor \frac{N}{2} \rfloor$ .

Without loss of generality and for the sake of parsimony, we consider hereafter the special case where the truncated two-dimensional SVD, derives from (5) when  $R = 1$ , i.e., when only the first two singular vectors (first *bimension*) are considered, i.e.

$$\mathbf{B} = \sum_{c, \tilde{c} \in \mathcal{C}, c < \tilde{c}} \left[ \mathbf{v}_1^{(c, \tilde{c})} \mathbf{v}_2^{(c, \tilde{c})\top} - \mathbf{v}_2^{(c, \tilde{c})} \mathbf{v}_1^{(c, \tilde{c})\top} \right] + \mathbf{\Xi}, \quad (7)$$

where,  $\mathbf{v}_1^{(c, \tilde{c})}$  and  $\mathbf{v}_2^{(c, \tilde{c})}$  are the two orthogonal vectors corresponding to the largest singular value  $\lambda_1^{(c, \tilde{c})}$ .

Therefore, plugging (7) into (1), the model can be formulated as

$$\mathbf{K} = \sum_{c, \tilde{c} \in \mathcal{C}, c < \tilde{c}} \left[ \mathbf{v}_1^{(c, \tilde{c})} \mathbf{v}_2^{(c, \tilde{c})\top} - \mathbf{v}_2^{(c, \tilde{c})} \mathbf{v}_1^{(c, \tilde{c})\top} \right] + \mathbf{E}, \quad (8)$$

subject to

$$u_{ic} \in \{0, 1\}, \quad (c = 1, \dots, C; \quad i = 1, \dots, N), \quad (9)$$

$$\sum_{c=1}^C u_{ic} = 1, \quad (i = 1, \dots, N), \quad (10)$$

$$\mathbf{v}_1^{(c, \tilde{c})\top} \mathbf{v}_2^{(c, \tilde{c})} = 0, \quad (c, \tilde{c} \in \mathcal{C}, c < \tilde{c}), \quad (11)$$

where the  $(N \times N)$  matrix  $\mathbf{E}$  in (8) is the error term that represents the part of  $\mathbf{K}$  not accounted for by the model. Constraints (9) and (10) qualify  $\mathbf{U}$  as a membership matrix, while (11) are constraints of orthogonality. Note that  $\mathbf{E}$  also incorporates the within part  $\mathbf{W}$  which is not modelled here.

Model (8) is fitted in the least-squares sense by minimising the following relative loss function

$$F\left(\mathbf{U}, \left\{\mathbf{v}_j^{(c, \tilde{c})}\right\}_{j=1,2, c, \tilde{c} \in \mathcal{C}, c < \tilde{c}}\right) = \frac{\left\|\mathbf{K} - \sum_{c, \tilde{c} \in \mathcal{C}, c < \tilde{c}} \left(\mathbf{v}_1^{(c, \tilde{c})} \mathbf{v}_2^{(c, \tilde{c})\top} - \mathbf{v}_2^{(c, \tilde{c})} \mathbf{v}_1^{(c, \tilde{c})\top}\right)\right\|^2}{\|\mathbf{K}\|^2}, \quad (12)$$

subject to the sets of constraints (9), (10), and (11).

In order to minimise (12), an ALS algorithm is proposed which iteratively updates each parameter while keeping all the others fixed as detailed in Sect. 4.

## 4 The ALS algorithm

In the general standard framework of ALS algorithms, an efficient algorithm is formulated and appropriately designed for fitting model (8).

The constrained problem of minimising (12) subject to (9), (10), and (11) can be solved by using an ALS algorithm which alternates between two main steps for updating  $\mathbf{v}_j^{(c, \tilde{c})}$  ( $j = 1, 2$  and  $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) and  $\mathbf{U}$  as follows:

*Step 0. Initialization.*

*Step 1. Updating  $\mathbf{v}_j^{(c, \tilde{c})}$  ( $j = 1, 2, c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ):* given  $\mathbf{U}$ , vectors  $\mathbf{v}_j^{(c, \tilde{c})}$  are estimated as solutions of constrained regression problems.

*Step 2. Updating  $\mathbf{U}$ :* given  $\mathbf{v}_j^{(c, \tilde{c})}$  ( $j = 1, 2$  and  $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ), membership matrix  $\mathbf{U}$  is updated in a row-wise fashion by solving assignment problems.

*Step 3. Stopping rule.*

Steps 1 to 3 are alternated and iterated until convergence. The loss function (12) 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 prevent from falling into local optima, the best solution is retained from a number of different (random or rational) starts. A detailed description of the steps of the algorithm implemented in MATLAB R2022 follows.

*Step 0. Initialization:* Choose a random or rational starting partition  $\widehat{\mathbf{U}}$  of the  $N$  objects into  $C$  non-empty clusters.

*Step 1. Updating  $\mathbf{v}_j^{(c, \tilde{c})}$  ( $j = 1, 2, c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ):*

Given the current partition  $\widehat{\mathbf{U}}$ , the estimation of the orthogonal vectors minimising (12) is obtained as the solution of the matrix fitting problem (8) subject to orthogonality constraints (11). It results to be a special case of a regression problem of reduced rank and it is known to come up to a truncated SVD (Ten Berge 2005). Therefore, given partition  $\widehat{\mathbf{U}}$ , the optimal estimates  $\widehat{\mathbf{v}}_j^{(c, \tilde{c})}$  ( $j = 1, 2$  and  $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) result to be proportional to the two singular vectors  $\widehat{\mathbf{p}}_j^{(c, \tilde{c})}$  ( $j = 1, 2$ ) associated with the largest

singular value  $\hat{\lambda}_1^{(c, \tilde{c})}$  of matrix  $\widehat{\mathbf{B}}^{(c, \tilde{c})}$

$$\hat{\mathbf{v}}_1^{(c, \tilde{c})} = \sqrt{\hat{\lambda}_1^{(c, \tilde{c})}} \hat{\mathbf{p}}_1^{(c, \tilde{c})}, \quad \hat{\mathbf{v}}_2^{(c, \tilde{c})} = \sqrt{\hat{\lambda}_1^{(c, \tilde{c})}} \hat{\mathbf{p}}_2^{(c, \tilde{c})}, \quad (c, \tilde{c} \in \mathcal{C}, c < \tilde{c}). \quad (13)$$

*Step 2. Updating U:* Given the current estimates of  $\hat{\mathbf{v}}_j^{(c, \tilde{c})}$  ( $j = 1, 2$  and  $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ), the updating of the membership matrix  $\widehat{\mathbf{U}}$  is done by solving  $N$  assignment problems which minimise the loss function (12). This problem is sequentially solved for the different rows of  $\mathbf{U}$  by taking  $\hat{u}_{it} = 1$  if column  $t$  attains  $F([u_{ih}], \cdot) = \min\{F([u_{ih}], \cdot) : h = 1, \dots, C\}$  and  $\hat{u}_{it} = 0$ , otherwise.

When updated the membership matrix, a check for avoiding possible empty clusters is carried out.

*Step 3. Stopping rule:* Compute the loss value  $F\left(\widehat{\mathbf{U}}, \left\{\hat{\mathbf{v}}_j^{(c, \tilde{c})}\right\}_{j=1,2, c, \tilde{c} \in \mathcal{C}, c < \tilde{c}}\right)$ , for the current estimates according to (12). When such updated values have decreased considerably (more than an arbitrary small convergence tolerance) the function value,  $\widehat{\mathbf{U}}$  and  $\hat{\mathbf{v}}_j^{(c, \tilde{c})}$  ( $j = 1, 2$ ) are updated once more according to Steps 1 and 2. Otherwise, the process is assumed to have converged.

**Remark 2** Note that the algorithm for fitting model (8) with more than one bimension ( $R > 1$ ) can be generally obtained by retaining  $2R$  singular vectors in Step 1 straightforwardly.

### 4.1 A computationally efficient estimation

An equivalent but much more computationally efficient form of SVD approximation can be considered in Step 1. Given partition  $\widehat{\mathbf{U}}$ , let  $G_c$  and  $G_{\tilde{c}}$  be a pair of clusters of sizes  $n_c$  and  $n_{\tilde{c}}$ , respectively.

Interestingly, since matrices  $\widehat{\mathbf{B}}^{(c, \tilde{c})}$  ( $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) are skew-symmetric off-diagonal block matrices, vectors  $\hat{\mathbf{v}}_1^{(c, \tilde{c})}$  and  $\hat{\mathbf{v}}_2^{(c, \tilde{c})}$  take the following form by construction

$$\hat{v}_{1i}^{(c, \tilde{c})} \begin{cases} \neq 0 & \text{if } i \in G_c \\ = 0 & \text{otherwise,} \end{cases} \quad \hat{v}_{2i}^{(c, \tilde{c})} \begin{cases} \neq 0 & \text{if } i \in G_{\tilde{c}} \\ = 0 & \text{otherwise,} \end{cases}$$

or

$$\hat{v}_{1i}^{(c, \tilde{c})} \begin{cases} \neq 0 & \text{if } i \in G_{\tilde{c}} \\ = 0 & \text{otherwise,} \end{cases} \quad \hat{v}_{2i}^{(c, \tilde{c})} \begin{cases} \neq 0 & \text{if } i \in G_c \\ = 0 & \text{otherwise,} \end{cases} \quad (14)$$

for  $i = 1, \dots, N$  and  $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ .

Due to (14), any estimated matrix  $\widehat{\mathbf{K}}^{(c, \tilde{c})}$  ( $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) results to be the  $(N \times N)$  skew-symmetric matrix of the imbalances between clusters  $G_c$  and  $G_{\tilde{c}}$ , i.e., the non-null entries correspond only to pairs of objects  $i \in G_c$  and  $j \in G_{\tilde{c}}$  ( $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ).

Let  $\hat{\beta}^{(c, \tilde{c})}$  denotes the submatrix of size  $(n_c \times n_{\tilde{c}})$  extracted from  $\hat{\mathbf{K}}^{(c, \tilde{c})}$  and containing only its rectangular block of the imbalances from  $G_c$  to  $G_{\tilde{c}}$  and let  $\hat{\boldsymbol{\pi}}^{(c, \tilde{c})}$  and  $\hat{\boldsymbol{\rho}}^{(c, \tilde{c})}$  be the first left and right singular vectors of sizes  $n_c$  and  $n_{\tilde{c}}$ , respectively, of the truncated SVD of  $\hat{\beta}^{(c, \tilde{c})}$ . Then, the very elements of the singular vectors  $\hat{\boldsymbol{\rho}}_1^{(c, \tilde{c})}$  and  $\hat{\boldsymbol{\rho}}_2^{(c, \tilde{c})}$  of  $\hat{\mathbf{B}}^{(c, \tilde{c})}$  in (13) are actually  $\hat{\boldsymbol{\pi}}^{(c, \tilde{c})}$  and  $\hat{\boldsymbol{\rho}}^{(c, \tilde{c})}$  respectively, and they can be written as

$$\hat{\rho}_{1i}^{(c, \tilde{c})} = \begin{cases} \hat{\pi}_h^{(c, \tilde{c})} & \text{if } \hat{u}_{ic} = 1 \\ 0 & \text{otherwise,} \end{cases} \quad \hat{\rho}_{2i}^{(c, \tilde{c})} = \begin{cases} \hat{\rho}_l^{(c, \tilde{c})} & \text{if } \hat{u}_{i\tilde{c}} = 1 \\ 0 & \text{otherwise,} \end{cases} \tag{15}$$

for  $h = 1, \dots, n_c, l = 1, \dots, n_{\tilde{c}}$  and  $i = 1, \dots, N$ . Vectors  $\hat{\mathbf{v}}_1^{(c, \tilde{c})}$  and  $\hat{\mathbf{v}}_2^{(c, \tilde{c})}$  are computed as in (13) accordingly.

Note that such a solution involves the SVD of  $\binom{C}{2}$  matrices of sizes  $(n_c \times n_{\tilde{c}})$ , for  $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$  which are generally much smaller than matrices  $\mathbf{B}^{(c, \tilde{c})}$  of size  $(N \times N)$ .

Thus, in order to speed up the algorithm, the estimation of  $\hat{\mathbf{v}}_j^{(c, \tilde{c})}$  ( $j = 1, 2$  and  $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) in (13) can be done in a more computationally efficient way by estimating their very elements  $\hat{\pi}_h^{(c, \tilde{c})}$  ( $h = 1, \dots, n_c$ ) and  $\hat{\rho}_l^{(c, \tilde{c})}$  ( $l = 1, \dots, n_{\tilde{c}}$ ), respectively, as follows

<pre> <b>for</b> <math>h = 1, \dots, N</math>   <b>if</b> <math>\hat{u}_{ic} = 1</math>     <math>\hat{v}_{1i}^{(c, \tilde{c})} = \sqrt{\hat{\lambda}_1^{(c, \tilde{c})}} \hat{\pi}_h^{(c, \tilde{c})}</math>     <math>h = h + 1</math>   <b>else</b>     <math>\hat{v}_{1i}^{(c, \tilde{c})} = 0</math>   <b>end</b> <b>end</b> </pre>	<pre> <b>for</b> <math>l = 1, \dots, N</math>   <b>if</b> <math>\hat{u}_{i\tilde{c}} = 1</math>     <math>\hat{v}_{2i}^{(c, \tilde{c})} = \sqrt{\hat{\lambda}_1^{(c, \tilde{c})}} \hat{\rho}_l^{(c, \tilde{c})}</math>     <math>l = l + 1</math>   <b>else</b>     <math>\hat{v}_{2i}^{(c, \tilde{c})} = 0</math>   <b>end</b> <b>end</b> </pre>
--	---

### 4.2 An equivalent form for the loss function

Due to SVD (Appendix 1), the relative loss function (12) is equivalent to

$$F(\mathbf{U}, \{\lambda_1^{(c, \tilde{c})}\}_{c, \tilde{c} \in \mathcal{C}, c < \tilde{c}}) = 1 - \frac{2 \sum_{c, \tilde{c} \in \mathcal{C}, c < \tilde{c}} (\lambda_1^{(c, \tilde{c})})^2}{\|\mathbf{K}\|^2} \tag{16}$$

In the algorithm the computation of the relative loss function (16) in Steps 2 and 3 can be profitably used, because it does not require to compute the singular vectors of size  $N$  as in (12) and it turns out to be computationally more efficient.

**Remark 3** It can be observed that the computational complexity of the algorithm is given by its two main steps.

- Given partition  $\mathbf{U}$ , Step 1 actually consists of  $\binom{C}{2}$  truncated SVDs of “small” matrices of size  $(n_c \times n_{\tilde{c}})$  ( $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ).
- Given the singular vectors, Step 2 is the standard allocation step of all  $k$ -means-type algorithms where the loss function is computed by using (16).

In order to provide details about the computational effort, the analysis of the computation time for the simulation study is reported in Sect. 5.3.

## 5 Simulation

In order to evaluate the performance of the model, a simulation study has been carried out on artificial data (Sects. 5.1-5.3). A comparison with the CLUSKEXT model (Vicari 2018), its closest least-squares-based competitor, is also reported in Sect. 5.4.

### 5.1 Simulation design and measures of performance

A number of skew-symmetric matrices have been generated from the true underlying model (8) by setting  $N = 20, 40$  objects and  $C = 2, 3, 4, 5$  clusters of approximately equal sizes.

Specifically, a random partition into  $C$  non-empty clusters has been drawn from a discrete uniform distribution so that any object is randomly assigned to a cluster with probability  $\frac{1}{C}$ . Then, vectors  $\mathbf{v}_1^{(c, \tilde{c})}$  and  $\mathbf{v}_2^{(c, \tilde{c})}$  ( $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) have been randomly generated by taking into account the special form (14). Specifically, the non-null components of  $\mathbf{v}_1^{(c, \tilde{c})}$  and  $\mathbf{v}_2^{(c, \tilde{c})}$  have been computed as in (13) by setting  $\lambda_1^{(c, \tilde{c})} = 1$  and generating vectors  $\mathbf{p}_1^{(c, \tilde{c})}, \mathbf{p}_2^{(c, \tilde{c})}$  ( $c, \tilde{c} \in \mathcal{C}, c < \tilde{c}$ ) from discrete uniform distributions in  $[1, 10]$ . Then, any true skew-symmetric off-diagonal block matrix  $\mathbf{K}^*$  has been computed as

$$\mathbf{K}^* = \sum_{c, \tilde{c} \in \mathcal{C}, c < \tilde{c}} \left[ \mathbf{v}_1^{(c, \tilde{c})} \mathbf{v}_2^{(c, \tilde{c})\top} - \mathbf{v}_2^{(c, \tilde{c})} \mathbf{v}_1^{(c, \tilde{c})\top} \right]$$

and then perturbed as follows

$$\mathbf{K} = \mathbf{K}^* + \sqrt{\delta} \mathbf{E},$$

where  $\delta$  has been set equal to 0.15, 0.25, 0.50, 0.75 to allow for different error levels; as for the error matrix  $\mathbf{E}$ , a matrix  $\tilde{\mathbf{E}}$  of size  $(N \times N)$  has been firstly generated from a standard normal distribution, and then it has been skew-symmetrised as  $\mathbf{E} = \tilde{\mathbf{E}} - \tilde{\mathbf{E}}^\top$  and rescaled to have the same sum of squares as the error-free data.

For each cell of the experiment, 100 data sets have been generated as follows:

- 3 (number of clusters  $C = 2, 3, 4$ )  $\times$  4 (error levels)=12 data sets of sample size  $N = 20$ ,

(b)  $4$  (number of clusters  $C = 2, 3, 4, 5$ )  $\times$   $4$  (error levels) = 16 data sets of sample size  $N = 40$ ,

for a total of 2800 data sets. For each data set the best solution in terms of loss function in 100 runs of the algorithm from different random starts has been retained so that the algorithm was run 280000 times in total.

The simulation study has been performed on an AMD Epyc 7452 processor and 2GB RAM.

In order to evaluate the performance of the algorithm, for each cell of the experimental design the following measures have been computed by averaging over the 100 data sets:

1. *ARI*: Adjusted Rand Index (Hubert and Arabie 1985) between true  $\mathbf{U}$  and fitted  $\hat{\mathbf{U}}$  membership matrices: it measures the degree of agreement between two partitions and takes its maximum value equals to 1 when the two partitions are coincident;
2.  $\%(ARI=1)$ : percentage of successes in recovering the true partitions, i.e. % of times where  $ARI=1$ ;
3.  $\#(ARI=1)$ : number of times where  $ARI=1$ ;
4.  $\#(ARI>0.85)$ : number of times where  $ARI>0.85$ ;
5. *LOSS*: relative loss function value (16): it takes values in  $[0, 1]$ ;
6. *TIME*: time per run (in seconds);
7. *# ITER*: number of iterations before convergence (tolerance value equal to  $10^{-5}$ );
8. *TCC*: Tucker's Congruence Coefficient (Tucker 1951): squared cosine of the angle between the subspaces spanned by the true  $\mathbf{v}_1^{(c, \tilde{c})}$  and  $\mathbf{v}_2^{(c, \tilde{c})}$  and the estimated  $\hat{\mathbf{v}}_1^{(c, \tilde{c})}$  and  $\hat{\mathbf{v}}_2^{(c, \tilde{c})}$  ( $c, \tilde{c} \in \mathcal{C}$ ,  $c < \tilde{c}$ ). It measures the degree of agreement between two subspaces and takes values in  $[0, 1]$  where the maximum value indicates that the two subspaces are coincident. Specifically, since there are  $\binom{C}{2}$  squared cosine values corresponding to the  $\binom{C}{2}$  off-diagonal blocks of any matrix  $\mathbf{K}$ , for each data set the squared cosine values have been averaged.

## 5.2 Simulation results from the model

The results of the simulation study are displayed in Tables 2 and 3 and Figs. 6, 7, 6, 8, 9, where the average measures of performance are reported for the two scenarios with  $N = 20$  and  $N = 40$ . Generally speaking, the tables show a good performance of the algorithm for both cluster (Fig. 6a) and subspace recovery (Fig. 6b), even when the error level is high and one can see that the performance dramatically drops only in the more complex settings in the presence of many clusters and very high error level, especially when the sample size is smaller.

To deepen the analysis of the performance of the algorithm, the average values of *ARI* and *TCC* for each cell of the experimental design can be analysed in more detail. Given the number of clusters  $C$ , the performance in terms of recovery of the true both partition and subspace improves with the sample size and decreases with the error level (Tables 2 and 3, Fig. 6a and b), as expected.



Moreover, as evident from the boxplots in Figs. 7 and 8, the ARI and TCC distributions exhibit an increasing variability as the level of error  $\delta$  and the number of groups  $C$  increase, but the measures generally maintain good values except for the case with a very high level of error and many clusters. Note that TCC has been used as a measure of how well the algorithm correctly reveals the underlying clustering subspace and  $\binom{C}{2}$  TCC values have been averaged for each cell of the experimental design: this affects the variability of its distribution which is generally greater than that of ARI (Fig. 8).

The trend of the loss values (Figs. 6c and 9) is consistent with both cluster and subspace recovery, so the means and variability of the distributions increase with error and decrease with sample size on average (Fig. 9). It can be observed that with the same sample size and error, the relative loss decrease with the number of clusters (Fig. 9): this is due to the fact that as  $C$  increases, the *between* blocks become larger in size and the residual *within* part of the data that remains unexplained gets smaller and smaller.

As for the scalability, in Tables 2 and 3 the values of the average run time (in seconds) and the number of iterations before convergence are reported. The number of iterations, while increasing with the error, remains low and does not change remarkably as  $C$  and  $N$  vary. As for the computation time per run, given the same number of clusters, the increasing error does not seem to provide substantial differences. Conversely, when the sample size doubles for the same number of clusters, the average run time is nearly doubled (Fig. 6d). Specifically, for  $N = 40$  the average run time is 2.77, 1.61 and 1.5 times higher than the case with  $N = 20$  for  $C = 2, 3, 4$ , respectively.

Further details on computation time emerge by inspecting Tables 2 and 3. We can

**Table 2** Simulation study: sample size equal to 20

N = 20							
C	$\delta$	ARI	%(ARI=1)	TCC	LOSS	TIME	#ITER
2	0.15	1	100	0.991	0.117	0.07	2.04
	0.25	1	100	0.984	0.182	0.07	2.07
	0.50	1	100	0.969	0.301	0.07	2.05
	0.75	0.998	99	0.952	0.384	0.08	2.19
3	0.15	1	100	0.982	0.105	0.37	2.50
	0.25	0.995	97	0.966	0.163	0.40	2.60
	0.50	0.936	69	0.898	0.264	0.46	3.25
	0.75	0.729	11	0.721	0.337	0.50	3.57
4	0.15	0.995	97	0.970	0.093	1.48	3.17
	0.25	0.961	75	0.928	0.143	1.77	3.77
	0.50	0.591	6	0.585	0.224	1.74	4.32
	0.75	0.328	0	0.340	0.265	1.79	4.29

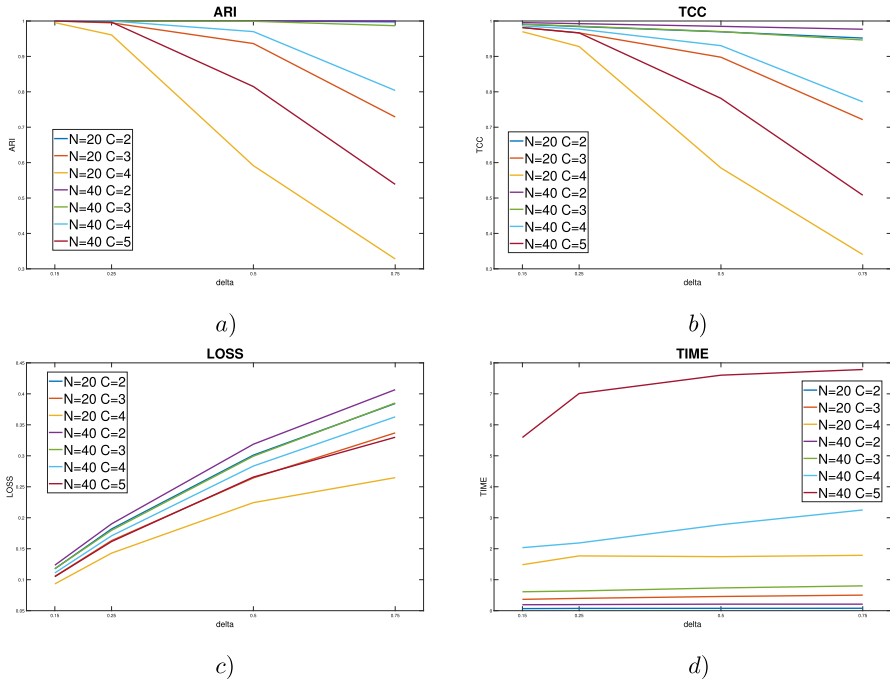
**Table 3** Simulation study: sample size equal to 40

N = 40							
C	$\delta$	ARI	%(ARI=1)	TCC	LOSS	TIME	#ITER
2	0.15	1	100	0.996	0.123	0.19	2.01
	0.25	1	100	0.992	0.190	0.20	2.03
	0.50	1	100	0.985	0.319	0.21	2.03
	0.75	1	100	0.977	0.407	0.21	2.05
3	0.15	1	100	0.991	0.118	0.61	2.32
	0.25	1	100	0.985	0.180	0.64	2.35
	0.50	0.999	99	0.970	0.299	0.73	2.70
	0.75	0.987	84	0.946	0.385	0.80	2.99
4	0.15	1	100	0.986	0.111	2.03	2.91
	0.25	1	100	0.977	0.171	2.19	3.03
	0.50	0.970	62	0.930	0.283	2.78	3.70
	0.75	0.804	7	0.772	0.363	3.25	4.56
5	0.15	1	100	0.981	0.105	5.59	3.42
	0.25	0.997	96	0.966	0.162	7.01	3.69
	0.50	0.815	5	0.781	0.266	7.60	5.06
	0.75	0.539	0	0.508	0.330	7.78	5.81

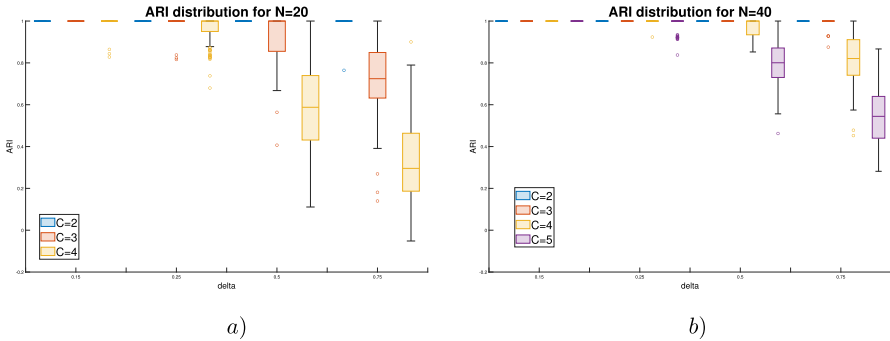
see that the average computation time per run increases with the complexity of the data (i.e., sample size, error level and number of clusters) and the divergence between small and large samples becomes more pronounced as the data becomes more complex in terms of number of clusters, especially when data are very highly perturbed. In detail, given  $N = 20$ , for  $C = 3$  and  $C = 4$  TIME increases by 3.45 and 12.75 times compared to the case with  $C = 2$ , respectively. When  $N = 40$ , for  $C = 3$  and  $C = 4$  TIME increases by 6.14 and 24.14 times compared to  $C = 2$ , respectively.

### 5.3 Local optima and number of starts

In order to analyse the stability of the solution and investigate about the sensitivity to local optima, further results are reported in Tables 4 and 5 which display the best ARI values over an increasing number of random starts (10, 30, 50, 100). Let us recall that at each run the algorithm starts from a random partition drawn from a discrete uniform distribution so that any object is randomly assigned to a cluster with probability  $1/C$ . Given the number of starts, both the average ARI between true and fitted partitions and the number of times where  $ARI = 1$  show an improvement on average as  $\delta$  decreases, as expected. Thus, a very good performance in terms of average ARI is already achieved when the optimal solution is retained over a few random starts when it generally becomes stable. Specifically, when the error is low ( $\delta = 0.15, 0.25$ ), the optimal solution is already achieved with 10 random starts. For high level of error

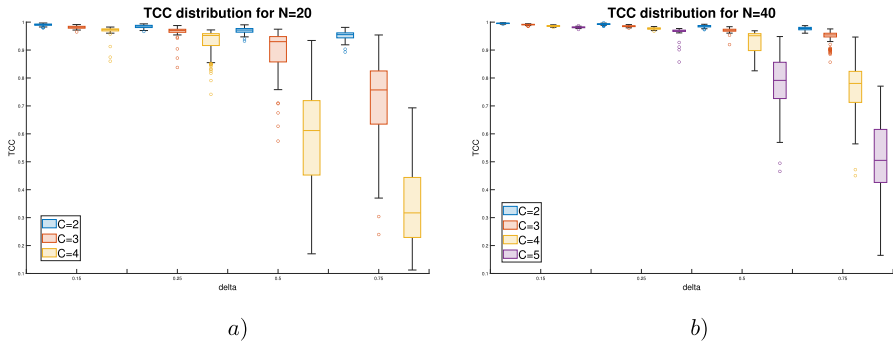


**Fig. 6** Measures of performance for  $C = 2, 3, 4, 5$  and  $N = 20, 40$ : **a** ARI; **b** TCC; **c** LOSS; **d** TIME (color figure online)

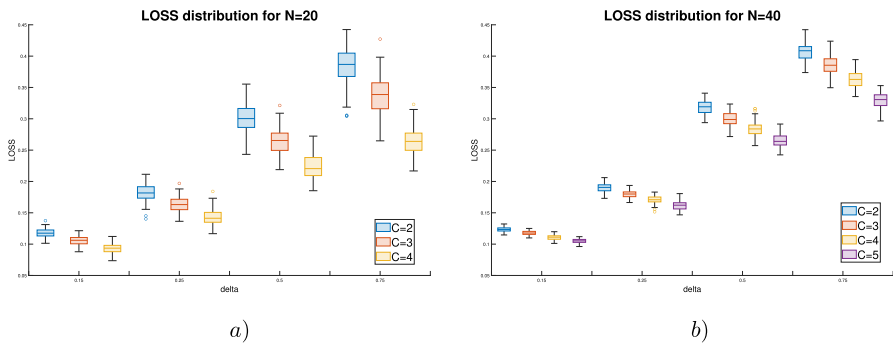


**Fig. 7** Box plots of the ARI distributions for increasing  $\delta$  and  $C = 2$  (blue),  $C = 3$  (red),  $C = 4$  (yellow),  $C = 5$  (violet) for: **a**  $N = 20$ ; **b**  $N = 40$  (color figure online)

( $\delta = 0.50$ ), small sample size and many clusters the chance to get the best solution, although increasing with the number of starts, remains low. When the error is very high ( $\delta = 0.75$ ) and with many clusters, the underlying clustering structure is masked and the optimal solution is never obtained for both sample sizes.



**Fig. 8** Box plots of the TCC distributions for increasing  $\delta$  and  $C = 2$  (blue),  $C = 3$  (red),  $C = 4$  (yellow),  $C = 5$  (violet) for: **a**  $N = 20$ ; **b**  $N = 40$  (color figure online)



**Fig. 9** Box plots of the LOSS distributions for increasing  $\delta$  and  $C = 2$  (blue),  $C = 3$  (red),  $C = 4$  (yellow),  $C = 5$  (violet) for: **a**  $N = 20$ ; **b**  $N = 40$

It is important to note that for  $C = 2$  the optimal solution is found with only 10 starting points, this is due to the fact that in this case only one block of the original exchanges between the two clusters have to be reconstructed and the optimal approximation is easier to achieve.

In order to further analyse the trend of the ARI as the number of starts increases, Tables 4 and 5 report the number of times where the ARI is higher than 0.85 which denotes a pretty good partition with not many misclassified objects. In the more complex cases, with many clusters and high error ( $\delta = 0.5$  and  $\delta = 0.75$ ), it can be observed that the quality of the resulting partition improves with the number of starts.

It is important to note that for  $C = 2$  and  $C = 3$  the optimal solution is always found with only 10 starting points, except for the case with a very high level of error where, however, the chance to get either the true ( $ARI = 1$ ) or at least a good partition ( $ARI > 0.85$ ) is still high.

All in all, we may observe that the local minima problem does not result to be generally crucial except for small datasets and many clusters where, in the presence of a high level of noise that can mask the existing clustering structure, the percentage

of times where the partitions are correctly recovered might need a large number of random starts for a more relevant improvement.

#### 5.4 Comparison with CLUSKEXT model

In order to make a comparison, the CLUSKEXT model (Vicari 2018) has been considered by fitting the unconstrained model (with no external information) which defines the direct least-squares competitor of our model. Let us recall that while our proposal aims at reconstructing the original between-cluster imbalances in low dimensions, the CLUSKEXT model estimates the average imbalances.

The CLUSKEXT model has been fitted to the same datasets generated for the simulation study and its performances in terms of ARI,  $\%(ARI=1)$  and loss values are reported in Tables 6 and 7 together with the measures from our between-cluster model to facilitate the comparison. The recovery trends are similar when compared with those described for our model: the cluster recovery gets worse when the amount of error and the true number of clusters increase.

Anyway, the average values of ARI and  $\%(ARI=1)$  are rather high only for the case with two clusters while they dramatically drop for the cases with more clusters even with small amount of error.

By comparing the performances of CLUSKEXT and our between-cluster model, it can be observed that the latter outperforms the former: for any cell of the experimental design, the average values of ARI and  $\%(ARI=1)$  for our model are never lower than the corresponding ones for CLUSKEXT.

In a few cases, the average loss values coincide or are almost the same and this often occurs in conditions where (essentially) perfect recoveries are obtained. In some other cases, the differences in the average values are much more pronounced and this holds when the level of error is high or very high and, above all, when  $C$  increases. All in all, we therefore observe that different partitions are discovered by fitting CLUSKEXT model and our between-cluster model, consistently with their different aims.

## 6 A real application: cola brand switching data

A real application of brand switching data derived from supermarket scanner data has been considered, in order to investigate how households change in buying cola soft drinks. The daily purchases of 15 different cola soft drinks were recorded for 488 US households over a period of 104 weeks from June 1991 to June 1993 (Bell and Lattin 1998).

In the original data rows indicate cola brands bought before and columns indicate cola brands that are currently bought; thus, changes are made from row to column products. In the view of analysing the directions of the brand choices and to take into account the information about the diagonal entries which measure the brand loyalty, a skew-symmetric matrix  $\mathbf{K} = (k_{ij})$  has been built from the original data  $\mathbf{D} = (d_{ij})$  by the following transformation:  $k_{ij} = (d_{ij} - d_{ji} + d_{ii} - d_{jj})/2$ , where the imbalance between  $i$  and  $j$  is corrected for the imbalance between the loyalties of  $j$  and  $i$  (Saito

**Table 4** Analysis of random starts ( $N = 20$ )

C	#starts	$\delta=0.15$			$\delta=0.25$			$\delta=0.50$			$\delta=0.75$		
		ARI	#ARI=1	#ARI>0.85	ARI	#ARI=1	#ARI>0.85	ARI	#ARI=1	#ARI>0.85	ARI	#ARI=1	#ARI>0.85
2	10	1	10	10	1	10	10	1	10	10	1	10	10
	30	1	30	30	1	30	30	1	30	30	1	30	30
	50	1	50	50	1	50	50	1	50	50	1	50	50
	100	1	100	100	1	100	100	1	100	100	0.998	99	99
3	10	1	10	10	1	10	10	1	7	7	0.740	2	3
	30	1	30	30	0.989	28	28	0.989	20	23	0.711	2	5
	50	1	50	50	0.993	48	48	0.993	32	37	0.703	5	11
	100	1	100	100	0.995	97	97	0.995	69	77	0.729	11	24
4	10	1	10	10	0.960	7	9	0.660	0	1	0.330	0	0
	30	0.989	28	28	0.962	22	27	0.631	2	6	0.360	0	1
	50	0.993	48	48	0.971	40	46	0.625	4	9	0.331	0	1
	100	0.995	97	98	0.961	75	88	0.624	6	14	0.328	0	1

**Table 5** Analysis of random starts ( $N = 40$ )

C	#starts	$\delta = 0.15$			$\delta = 0.25$			$\delta = 0.50$			$\delta = 0.75$		
		ARI	#ARI=1	#ARI>0.85	ARI	#ARI=1	#ARI>0.85	ARI	#ARI=1	#ARI>0.85	ARI	#ARI=1	#ARI>0.85
2	10	1	10	10	1	10	10	1	10	10	1	10	10
	30	1	30	30	1	30	30	1	30	30	1	30	30
	50	1	50	50	1	50	50	1	50	50	1	50	50
	100	1	100	100	1	100	100	1	100	100	1	100	100
3	10	1	10	10	1	10	10	1	10	10	0.983	8	10
	30	1	30	30	1	30	30	0.997	29	30	0.994	28	30
	50	1	50	50	1	50	50	0.998	49	50	0.989	43	50
	100	1	100	100	1	100	100	0.999	99	100	0.987	84	99
4	10	1	10	10	1	10	10	0.978	8	10	0.766	1	3
	30	1	30	30	1	30	30	0.978	22	30	0.804	3	14
	50	1	50	50	1	50	50	0.973	32	50	0.806	4	22
	100	1	100	100	1	100	100	0.970	62	100	0.804	7	43
5	10	1	10	10	1	10	10	0.809	1	4	0.561	0	0
	30	1	30	30	1	30	30	0.815	1	13	0.567	0	1
	50	1	50	50	0.996	48	50	0.831	4	26	0.541	0	1
	100	1	100	100	0.997	96	100	0.815	5	46	0.539	0	1

**Table 6** Comparison with CLUSKEXT

N=20							
C	$\delta$	Between-cluster model			CLUSKEXT model		
		ARI	%(ARI=1)	LOSS	ARI	%(ARI=1)	LOSS
2	0.15	1	100	0.117	1	100	0.117
	0.25	1	100	0.182	0.98	98	0.183
	0.50	1	100	0.301	0.767	76	0.3
	0.75	0.998	99	0.384	0.517	50	0.381
3	0.15	1	100	0.105	0.441	3	0.116
	0.25	0.995	97	0.163	0.449	4	0.178
	0.50	0.936	69	0.264	0.422	1	0.286
	0.75	0.729	11	0.337	0.383	0	0.359
4	0.15	0.995	97	0.093	0.361	0	0.153
	0.25	0.961	75	0.143	0.356	0	0.202
	0.50	0.591	6	0.224	0.331	0	0.295
	0.75	0.328	0	0.265	0.294	0	0.364

Simulation study: sample size equal to 20

and Yadohisa 2005). This can be appropriate in the analysis of brand switching where the direction of the switches is often related to the degree of brand loyalty as it usually happens where the diagonal entries are large in comparison with the off-diagonal ones.

Here, we are interested in studying the asymmetry in changing the purchased brands. The 15 colas are: Coke decaf (*CD*), Coke diet decaf (*CdD*), Pepsi diet decaf (*PdD*), Pepsi decaf (*PD*), Canfield (*Can*), Coke (*C*), Coke classic (*CCL*), Coke diet (*Cd*), Pepsi diet (*Pd*), RC diet (*RCd*), Rite diet (*Rd*), Pepsi (*P*), Private label (*Pr*), RC (*RC*), Wildwood (*Wil*).

The proposed model is fitted by varying  $C = 1, \dots, 6$  and, from the analysis of the scree plot of the model fit (Fig. 10), the partition into three clusters has been chosen (goodness-of-fit = 97.73%) that corresponds to the the elbow of the scree plot.

The resulting partition of the brands is  $G_1 = \{CD, PdD, PD, Can, C, RCd, Wil\}$ ,  $G_2 = \{CCL, Cd, P\}$ ,  $G_3 = \{CdD, Pd, Rd, Pr, RC\}$ . In Figs. 11, 12, 13 the plots of the colas in the planes of the optimal vectors  $\hat{v}_1^{(c, \tilde{c})}$  and  $\hat{v}_2^{(c, \tilde{c})}$  ( $c, \tilde{c} \in \{1, 2, 3\}$ ,  $c < \tilde{c}$ ) are reported and the relations between cluster  $G_1$  (Fig. 11),  $G_2$  (Fig. 12) and  $G_3$  (Fig. 13), with all the others are visualised, respectively, and different markers denote different clusters: circles for objects in  $G_1$ , squares for objects in  $G_2$  and stars for objects in  $G_3$ . Moreover, the arrows in different colours represent the pairwise directions of the switches between clusters: red for  $(G_1, G_2)$ , blue for  $(G_1, G_3)$  and green for  $(G_2, G_3)$ . According to the geometrical interpretation of the SVD of skew-symmetric matrices (Constantine and Gower 1978), Fig. 11 shows an anticlockwise direction from  $G_1$  to  $G_2$  and  $G_3$ , i.e., since  $G_1$  has in-flows from  $G_2$  and  $G_3$ ,  $G_1$  is a *destination* cluster. Similarly, the plot in Fig. 12 displays a clockwise direction from

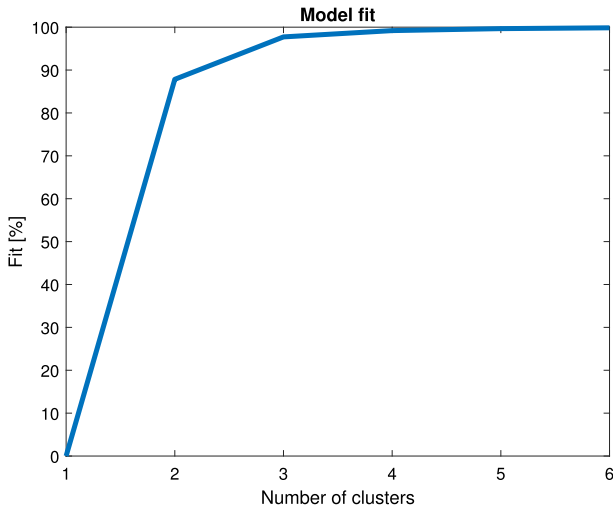


**Table 7** Comparison with CLUSKEXT

$N = 40$

C	$\delta$	Between-cluster model			CLUSKEXT model		
		ARI	%(ARI=1)	LOSS	ARI	%(ARI=1)	LOSS
2	0.15	1	100	0.123	1	100	0.124
	0.25	1	100	0.190	1	100	0.19
	0.50	1	100	0.319	0.99	99	0.319
	0.75	1	100	0.407	0.799	80	0.407
3	0.15	1	100	0.118	0.433	0	0.123
	0.25	1	100	0.180	0.436	1	0.187
	0.50	0.999	99	0.299	0.438	0	0.309
	0.75	0.987	84	0.385	0.442	0	0.396
4	0.15	1	100	0.111	0.303	0	0.158
	0.25	1	100	0.171	0.293	0	0.221
	0.50	0.970	62	0.283	0.377	0	0.330
	0.75	0.804	7	0.363	0.391	0	0.409
5	0.15	1	100	0.105	0.381	0	0.181
	0.25	0.997	96	0.162	0.346	0	0.241
	0.50	0.815	5	0.266	0.338	0	0.344
	0.75	0.539	0	0.330	0.323	0	0.419

Simulation study: sample size equal to 40



**Fig. 10** Cola data. Scree plot of the percentage fit

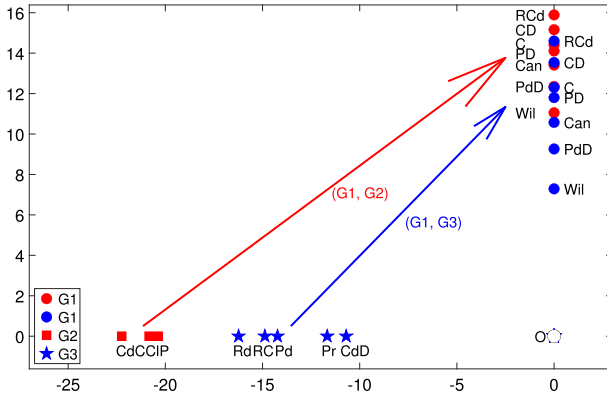


Fig. 11 *Cola data*. Scatter plot of the cola brands for cluster  $G_1$ . Directions of the switches between clusters are represented by arrows

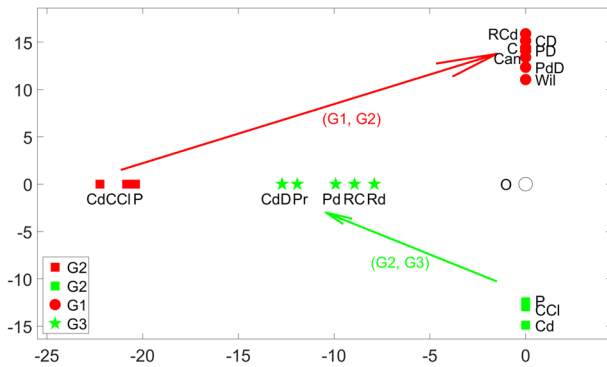


Fig. 12 *Cola data*. Scatter plot of the cola brands for cluster  $G_2$ . Directions of the switches between clusters are represented by arrows

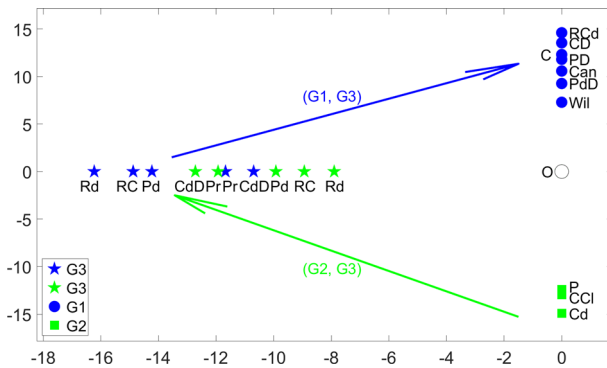
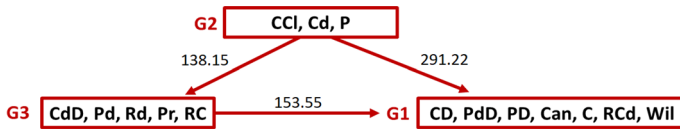


Fig. 13 *Cola data*. Scatter plot of the cola brands for cluster  $G_3$ . Directions of the switches between clusters are represented by arrows



**Fig. 14** Cola data. Graph of the clustering results where the values on the arrows represent the average brand switches between clusters

$G_2$  to both  $G_1$  and  $G_3$ , i.e.,  $G_2$  is an *origin* cluster, because it has out-flows towards  $G_1$  and  $G_3$ . Finally, in-flows from  $G_2$  and out-flows directed towards  $G_1$  qualify  $G_3$  as both a destination cluster from  $G_2$  and an origin cluster towards  $G_1$ .

Moreover, we recall that the area of the triangle formed by any pair of brands in different clusters and the origin is proportional to the amount of the imbalance between such pair of brands. Specifically, the areas of all triangles between brands in  $G_1 = \{CD, PdD, PD, Can, C, RCd, Wil\}$  and  $G_2 = \{CCl, Cd, P\}$  are larger than those between brands in  $G_2$  and  $G_3 = \{CdD, Pd, Rd, Pr, RC\}$  (Fig. 12), i.e., the switching from  $G_2$  to  $G_1$  is always greater than the switching from  $G_2$  to  $G_3$ . Similarly, switches towards brands in  $G_1$  are generally higher when coming from  $G_2$  rather than  $G_3$  (Fig. 11) and the switches from brands in  $G_3$  to brands in  $G_1$  are almost all higher than those from  $G_2$  to  $G_3$  (Fig. 13)

Finally, the results are summarized in Fig. 14 so that the origin/destination clusters are represented by arrowed lines according to what emerges from Figs. 11, 12, 13: households tend to switch their purchases from brands in cluster  $G_2$  to brands in clusters  $G_3$  and  $G_1$ . Specifically, from the values on the arrows in Fig. 14 which are computed as the average estimated switches between clusters, it results that households tend to switch mainly from the most popular brands such as Coke classic, diet Coke and Pepsi in favour to minor brand.

From the analysis of the cola features *diet/non-diet* and *caff/Decaf*, it can be observed that diet colas tend to be switched, while Decaf colas tend to represent the target of the switches, which is consistent with the results from the four-cluster solution of CLUSKEXT in Vicari (2018). Thus, the switches between cola brands can also be interpreted with respect to their features and using only a few clusters.

## 7 Concluding remarks

In this paper a novel clustering model oriented to exploit and convey the asymmetric relationships between objects is proposed. The model is mainly focused on the reconstruction of the pairwise imbalances between objects which, as a by-product, allows to account for the between-cluster effects and identify origin/destination clusters of objects.

The main strength of the proposal lies in the use of the SVD for skew-symmetric matrices, not only for its simplicity but also for the possibility of providing a graphical representation of the clustering results that are interpretable in terms of directed exchanges between clusters: clusters are identified as origins and/or destinations of the exchanges. The graphical representation is also able to visualise: (1) the amount of

the imbalance between any pair of objects belonging to different clusters as the area of the triangle formed by the objects and the origin; (2) the direction of the exchange.

The special form of the SVD of skew-symmetric matrices also allows to provide an efficient ALS algorithm specifically designed for the estimation process.

The method has been analysed thanks to an extensive simulation study which shows the good performance of the proposal and its ability in recovering the underlying clustering structure even in the presence of a high level of error and also in comparison with an existing method. However, as observed by one referee, a limitation of the numerical simulation could be given by the choices for the number of objects which is not large enough when the number of clusters is high. Therefore, it may be difficult to fully evaluate the effects of the number of clusters on the simulation results because the results of the cluster sizes are added to this effect, so this issue may deserve further future investigation.

A real-life application of brand switching data has been also presented to highlight the utility and simplicity of the cluster interpretation in terms of origin/destination clusters by using the graphical representation.

Further methodological developments may concern the modelling of the within component of the skew-symmetries which remains unexplained here in order to possibly reconstruct the exchanges within clusters.

Moreover, further insights may regard the analysis of the performance of the method in different real data applications to further investigate its capability and utility in various domains.

**Funding** Open access funding provided by Università degli Studi di Roma La Sapienza within the CRUI-CARE Agreement.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## A singular value decomposition of a skew-symmetric matrix

Let us recall the special form of the Singular Value Decomposition (SVD) of any skew-symmetric matrix (Gower 2018; Bove et al. 2021).

Let  $\mathbf{A} = (a_{ij})$  be an  $(N \times N)$  skew-symmetric matrix, where  $a_{ij} = -a_{ji}$  represents the imbalance between objects  $i$  and  $j$  ( $i, j = 1, \dots, N$ ).

Any skew-symmetric matrix can be written by using the full SVD as follows

$$\mathbf{A} = \mathbf{P}\mathbf{A}\mathbf{Q}^{\top} = \mathbf{P}\mathbf{A}\mathbf{J}\mathbf{P}^{\top}, \quad (\text{A1})$$

where

- $\mathbf{P}$  and  $\mathbf{Q}$  are the  $(N \times N)$  matrices of the left and right singular vectors, respectively;
- $\mathbf{\Lambda} = \text{diag}(\lambda_1, \lambda_1, \lambda_2, \lambda_2, \dots, \lambda_{[N/2]}, \lambda_{[N/2]})$  is the diagonal matrix of the singular values, where  $[\frac{N}{2}]$  denotes the integer part of  $\frac{N}{2}$ ;
- $\mathbf{J}$  is an  $(N \times N)$  block diagonal matrix with matrices  $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$  along its diagonal (if  $N$  is odd, the last element of  $\mathbf{J}$  is set to 1 to maintain orthogonality);

see (Bove et al. 2021; Gower 2018) for further details.

Note that singular values come in pairs and if  $N$  is odd the last one is zero. The equivalence in (A1) is due to the form of the skew-symmetric matrices so that to compute the SVD of  $\mathbf{A}$  it is sufficient to determine matrices  $\mathbf{P}$  and  $\mathbf{\Lambda}$ . Specifically, the presence of matrix  $\mathbf{J}$  in (A1) makes the pairs of the right singular vectors corresponding to the same singular value be a permutation and reflection of the corresponding left singular vectors, i.e., for example, the first pair of the left singular vectors  $\mathbf{p}_1, \mathbf{p}_2$  corresponds to the first pair of the right singular vectors, i.e.,  $\mathbf{q}_1 = \mathbf{p}_2, \mathbf{q}_2 = -\mathbf{p}_1$ . The pair  $(\mathbf{p}_n, \mathbf{q}_n)$  for  $n = 1, \dots, [N/2]$  is called the  $n$ -th *bimension*.

The full SVD of  $\mathbf{A}$  can be written as

$$\mathbf{A} = \sum_{n=1}^{[\frac{N}{2}]} \lambda_n (\mathbf{p}_{2n-1} \mathbf{p}_{2n}^\top - \mathbf{p}_{2n} \mathbf{p}_{2n-1}^\top),$$

which in scalar notation is

$$\begin{aligned} a_{ij} &= \lambda_1(p_{1i} p_{2j} - p_{1j} p_{2i}) + \lambda_2(p_{3i} p_{4j} - p_{3j} p_{4i}) + \dots + \\ &\quad + \lambda_{[N/2]}(p_{(N-1)i} p_{Nj} - p_{(N-1)j} p_{Ni}) = \\ &= \sum_{n=1}^{[\frac{N}{2}]} \lambda_n (p_{(2n-1)i} p_{(2n)j} - p_{(2n-1)j} p_{(2n)i}), \quad (i, j = 1, \dots, N), \end{aligned}$$

where  $p_{(2n-1)i}$  denotes the  $i$ -th element of the singular vector  $\mathbf{p}_{2n-1}$  of  $\mathbf{A}$  (similarly for  $p_{(2n)j}$ ),  $\lambda_n$  is the  $n$ -th singular value of  $\mathbf{A}$ .

The truncated form of the SVD of size  $2R$  can be written as

$$\mathbf{A} = \mathbf{P}_{(R)} \mathbf{\Lambda}_{(R)} \mathbf{J} \mathbf{P}_{(R)}^\top + \mathbf{E}, \quad R \leq \left\lceil \frac{N}{2} \right\rceil \tag{A2}$$

where  $\mathbf{P}_{(R)}$  denotes the  $(N \times 2R)$  matrix containing the first  $2R$  columns of  $\mathbf{P}$ ,  $\mathbf{\Lambda}_{(R)}$  the diagonal  $(2R \times 2R)$  matrix with elements  $\lambda_1, \lambda_1, \dots, \lambda_R, \lambda_R$ , and  $\mathbf{E}$  is the residual matrix from the full SVD in (A1). In scalar notation the expression (A2) can be written as

$$a_{ij} = \sum_{n=1}^R \lambda_n (p_{(2n-1)i} p_{(2n)j} - p_{(2n-1)j} p_{(2n)i}) + \epsilon_{ij}, \quad (i, j = 1, \dots, N),$$

where  $\epsilon_{ij}$  represents the residual term.

Finally, note that  $\mathbf{P}_{(R)}\mathbf{A}_{(R)}\mathbf{J}\mathbf{P}_{(R)}^\top$  in (A2) is the best approximation of rank  $2R$  of matrix  $\mathbf{A}$  in the least-squares sense and the following equivalence holds (see (Ten Berge 2005; Lütkepohl 1997))

$$\left\| \mathbf{A} - \mathbf{P}_{(R)}\mathbf{A}_{(R)}\mathbf{J}\mathbf{P}_{(R)}^\top \right\|^2 = 1 - \frac{2 \sum_{n=1}^R \lambda_n^2}{\|\mathbf{A}\|}.$$

## References

- Bell DR, Lattin JM (1998) Shopping behavior and consumer preference for store price format: why “large basket” shoppers prefer EDLP. *Mark Sci* 17:66–88
- Bove G, Okada A (2018) Methods for the analysis of asymmetric pairwise relationships. *Adv Data Anal Classif* 12:5–31
- Bove G, Okada A, Vicari D (2021) *Methods for the analysis of asymmetric proximity data*. Springer, Singapore
- Bove G, Vicari D (2023) Graphical analysis and clustering of asymmetric proximities. In: Okada A, Shigemasa K, Yoshino R, Yokoyama S (eds) *Facets of behaviormetrics*. Behaviormetrics: quantitative approaches to human behavior. Springer, Singapore, pp 23–36
- Constantine AG, Gower JC (1978) Graphic representations of asymmetric matrices. *Appl Stat* 27:297–304
- Eckart C, Young G (1936) The approximation of one matrix by another of lower rank. *Psychometrika* 1:211–218
- Gower JC (1977) The analysis of asymmetry and orthogonality. In: Cutsem B (ed) *Barra JR, Brodeau F, Romier G, Van. Recent Developments in Statistics*. North Holland, Amsterdam, pp 109–123
- Gower JC (2018) Skew symmetry in retrospect. *Adv Data Anal Classif* 12:33–41
- Hubert L, Arabie P (1985) Comparing partitions. *J Classif* 2:193–218
- Lütkepohl H (1997) *Handbook of matrices*. Wiley, New York
- Olszewski D (2012) K-means clustering of asymmetric data. In: Corchado E et al (eds) *Hybrid Artificial Intelligent Systems 2012, Part I, vol 7208*. Lecture Notes in Computer Science. Springer, Berlin, pp 243–254
- Olszewski D, Ster B (2014) Asymmetric clustering using the alpha-beta divergence. *Pattern Recognit* 47(5):2031–2041
- Rocci R, Bove G (2002) Rotation techniques in asymmetric multidimensional scaling. *J Comput Graph Stat* 11:405–419
- Saito T, Yadohisa H (2005) *Data analysis of asymmetric structures*. Advanced Approaches in Computational Statistics. Marcel Dekker, New York
- Ten Berge JMF (2005) *Least squares optimization in multivariate analysis*. DSWO Press, Leiden
- Tucker L. R (1951) A method for synthesis of factor analysis studies. Personnel Research Section Report No.984, Washington D.C.: Department of the Army
- Vicari D (2014) Classification of asymmetric proximity data. *J Classif* 31(3):386–420
- Vicari D (2018) CLUSKEXT: CLUstering model for SKew-symmetric data including EXTernal information. *Adv Data Anal Classif* 12:43–64
- Vicari D (2020) Modeling asymmetric exchanges between clusters. In: Imaizumi T, Nakayama A, Yokoyama S (eds) *Advanced studies in behaviormetrics and data science*. behaviormetrics: quantitative approaches to human behavior, vol 5. Springer, Singapore, pp 297–313
- Zielman B, Heiser WJ (1996) Models for asymmetric proximities. *Br J Math Stat Psychol* 49:127–146

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.