Fuzzy clustering with spatial-temporal information

Pierpaolo D'Urso[1,*], Livia De Giovanni[b], Marta Disegna[c], Riccardo Massari[1]

[a]*Department of Social Sciences and Economics, Sapienza University of Roma, P.le Aldo Moro 5, 00185 Roma, Italy*

[b]*Department of Political Sciences, LUISS Guido Carli, Rome, Italy*

[c]*Accounting, Finance & Economics Department, Faculty of Management, Bournemouth University, 89 Holdenhurst Road, Bournemouth, BH8 8EB, United Kingdom*

Abstract

Clustering geographical units based on a set of quantitative features observed at several time occasions requires to deal with the complexity of both space and time information. In particular, one should consider (1) the spatial nature of the units to be clustered, (2) the characteristics of the space of multivariate time trajectories, and (3) the uncertainty related to the assignment of a geographical unit to a given cluster on the basis of the above complex features. This paper discusses a novel spatially constrained multivariate time series clustering for units characterised by different levels of spatial proximity. In particular, the Fuzzy Partitioning Around Medoids algorithm with Dynamic Time Warping dissimilarity measure and spatial penalization terms is applied to classify multivariate Spatial-Temporal series. The clustering method has been theoretically presented and discussed using both simulated and real data, highlighting its main features. In particular, the capability of embedding different levels of proximity among units, and the ability of considering time series with different length.

Keywords: Fuzzy Clustering, Partitioning Around Medoids, Time Series, Spatial information, Dynamic Time Warping, Tourism, Multilevel spatial proximity

1. Introduction

As Caiado et al. (2015) highlights, the (1) model- (2) feature- and (3) observation-based approaches are the main methodological veins developed in the past to aggregate units characterised by similar behaviour across time (for more details, see also Warren Liao, 2005; Caiado et al., 2015; D'Urso et al., 2016).

The idea behind the model-based clustering algorithms is to find the best mathematical/statistical model able to describe given time-varying data. The clustering is then performed on the parameter estimates (or on the residuals) of the fitted models (see, e.g., Piccolo, 1990; Maharaj, 1996; Garcia-Escudero & Gordaliza, 1999; Kalpakis et al., 2001; James & Sugar, 2003; Alonso & Maharaj, 2006; Caiado & Crato, 2010; Otranto, 2010; D'Urso et al., 2013b,a, 2016; D'Urso et al., 2017). Examples of model-based fuzzy clustering algorithms for univariate time series can be found in D'Urso et al. (2013a,b).

*Corresponding author

Email addresses: pierpaolo.durso@uniroma1.it (Pierpaolo D'Urso), ldegiovanni@luiss.it (Livia De Giovanni), disegnam@bournemouth.ac.uk (Marta Disegna), Riccardo.massari@uniroma1.it (Riccardo Massari)

Following the feature-based clustering approach, time series are clustered according to one of their specific features, such as the autocorrelation function (ACF), the periodogram, the density function or the wavelet information (see, e.g., Alonso & Maharaj, 2006; Caiado et al., 2006, 2009; D'Urso & Maharaj, 2009; Maharaj & D'Urso, 2010, 2011; D'Urso & Maharaj, 2012; D'Urso et al., 2014; Lafuente-Rego & Vilar, 2016; Vilar et al., 2017). In the fuzzy clustering framework, both univariate and multivariate time series wavelet features have been considered in Maharaj et al. (2010) and D'Urso & Maharaj (2012), while frequency domains of univariate time series have been taken into account in Maharaj & D'Urso (2011).

Observed time series, or suitable transformations, are instead the segmentation data used in the observation-based approach (see, e.g., D'Urso, 2005a; Coppi et al., 2010, and references therein). In the last decade, different fuzzy clustering algorithms have been proposed for both univariate and multivariate time series (see, e.g., Coppi & D'Urso, 2002, 2003, 2006; D'Urso, 2005b; D'Urso et al., 2015, 2016; D'Urso et al., 2017; D'Urso et al., 2017; Vilar et al., 2017).

Similarly, different methods have been suggested in the clustering literature to discover spatial patterns for different kind of spatial units, e.g., urban areas or image pixels. The main challenge these methods deal with is the identification of an appropriate algorithm to capture both spatial dependence and spatial heterogeneity. Following the categorisation suggested by Caiado et al. (2015), Fouedjio (2016) classifies clustering of spatial data into four main approaches: (1) non-spatial clustering with geographical coordinates as additional variables; (2) non-spatial clustering based on a spatial dissimilarity measure; (3) spatially constrained clustering; (4) model-based clustering. An example of spatially constrained fuzzy algorithm for urban areas is provided by Di Nola et al. (2000). Examples of applications for image pixels segmentation can be found in Tolias & Panas (1998a,b); Pham & Prince (1999); Liew et al. (2000, 2003); Pham (2001); Liew et al. (2003); Chuang et al. (2006).

A fifth approach worth of notice consists in including a spatial penalty term in the objective function of the clustering method, as suggested by Pham (2001). While this proposal has been introduced for solving image segmentation problem, the idea beyond can be easily extended to the clustering of geographical areas (Coppi et al., 2010).

When time information are available for space unit, the spatial time data array is a three-way data array (i.e. arrays of the type: spatial objects $\times$ variables $\times$ occasions). The spatial time data array $\mathbf{X}$ can be reduced to a *bi*-dimensional array by combining two of the three dimensions on the rows and assigning the remaining dimension to the columns (Krishnapuram & Freg, 1992; Shekhar et al., 2015). This dimensionality reduction allows for the classification of units by means of a traditional clustering technique at the expense of information loss. To overcome this drawback, several clustering for spatial-temporal series have been suggested in the literature. Following Disegna et al. (2017), clustering of spatial-temporal series can be classified into: (i) non-spatial time series clustering based on a spatial dissimilarity measure (Izakian et al., 2013); (ii) density-based clustering (Ester et al., 1996; Wang et al., 2006; Birant & Kut, 2007; Ienco & Bordogna, 2016; Xie et al., 2016); (iii) model-based clustering (Basford & McLachlan, 1985; Viroli, 2011; Torabi, 2014, 2016; Disegna et al., 2017); (iv) spatially constrained time series clustering (Hu & Sung, 2006; Coppi et al., 2010; Gao & Yu, 2016). Three-way data arrays have also been analysed by means of several fuzzy clustering algorithms (see, e.g., Sato & Sato, 1994; Sato et al., 1997; Gordon & Vichi, 2001; D'Urso, 2004, 2005a; Coppi et al., 2010). As for space data, Coppi et al. (2010) proposed the inclusion of the spatial penalty term in the objective function of a fuzzy clustering algorithm for spatial-temporal data too. The aim of this term is to reduce the membership degrees of all units contiguous to the generic i-th unit computed in all clusters but the c-th cluster to which the i-th unit belongs (Coppi et al., 2010).

In this study a generalisation of the fuzzy clustering algorithm with spatial penalization introduced by Coppi et al. (2010) is proposed. In particular, the innovation is threefold: firstly, we suggest to substitute the Euclidean distance with the Dynamic Time Warping (DTW) dissimilarity measure; secondly, we extend the Coppi et al. (2010)'s algorithm to the case in which data are characterised by different sources of spatial information; thirdly, a measure of spatial autocorrelation, the Fuzzy Moran (FM)'s index, is defined to study the autocorrelation of the final imprecise partition when several spatial penalty terms are considered.

The DTW dissimilarity measure has been selected instead of other more traditional distance measures, such as the well known Euclidean distance, mainly for its flexibility, the possibility to simultaneously consider both intensity and dynamic existing between time series, and thanks to its ability to compute distance among multivariate time series not necessarily of the same length.

The necessity to consider more than one spatial penalty term in the clustering algorithm is motivated by practical case studies in which units are characterised by different levels, or concepts, of proximity. For instance, European region are classified into three levels of Nomenclature of Territorial Units for Statistics (NUTS) geography classification and any clustering analysis of European cities should take into consideration these three levels.

Therefore, the Dynamic Time Warping Fuzzy C-Medoids for Spatial-Temporal Trajectories (DTW-FCMd-STT) clustering algorithm with penalty terms is proposed and described in this manuscript.

The paper is structured as follows: in section 2 the suggested algorithm is described and discussed in depth; in section 3 different simulated case studies are presented in order to show the main features of the algorithm; in section 4 the methodology is illustrated by analysing real data describing the behaviour of the tourism flows in a destination, i.e. spatial region. section 5 concludes.

2. The methodology

The starting point is represented by a spatial time data array (three-way data array), algebraically formalised as (D'Urso, 2000, 2004, 2005a):

$$\mathbf{X} \equiv \{x_{ijt} : i = 1, \ldots, I;\ j = 1, \ldots, J;\ t = 1, \ldots, T\} \tag{1}$$

where i indicates the generic unit (geographical area or region), j the variable, and t the generic time; x_{ijt} is the value of the j-th variable observed for the i-th unit at time t.

Notice that the time data array $\mathbf{X}$ can be synthetically represented by means of a *bi*-dimensional matrix combining two of the three indices i, j, t on the rows and assigning the remaining index to the columns. For instance, the time data array can be defined as the set of *bi*-dimensional matrices $\mathbf{X}_i, \mathbf{X}_t$, or $\mathbf{X}_j$ as follows:

$$\begin{aligned}
\mathbf{X}_i &\equiv \{x_{ijt} : \; j = 1, \ldots, J; \; t = 1, \ldots, T\} \\
\mathbf{X}_t &\equiv \{x_{ijt} : \; i = 1, \ldots, I; \; j = 1, \ldots, J\} \\
\mathbf{X}_j &\equiv \{x_{ijt} : \; i = 1, \ldots, I; \; t = 1, \ldots, T\}.
\end{aligned}$$

We also assume to have K additional pieces of information on spatial location of each units in relation with the others, i.e., K different levels of spatial proximity. Each level of proximity is defined by a $(I \times I)$ symmetric data matrix $\mathbf{P}_k$ $(k = 1, \ldots, K)$, whose generic entry $p_{kii'}$ is a measure of a particular definition of spatial proximity between the i-th and i'-th units $(i, i' = 1, \ldots, I)$, where $0 \leq p_{kii'} \leq 1$ and $p_{kii} = 0$. For instance, $p_{kii'} = 1$ if the two areas are contiguous, $p_{kii'} = 0$ otherwise. Alternatively, $p_{kii'}$ could be inversely proportional to the geographic distance between i and i'. We will further illustrate different kind of proximity matrix in section 2.2.

Figure 1 graphically represents the bundle of available information and the dimensions of the data array typically used in spatial-temporal analysis.

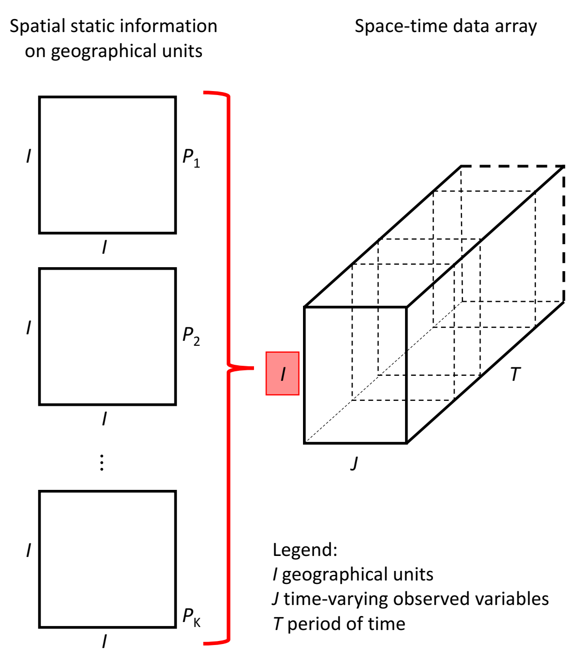


Figure 1: Spatial-temporal data array

For classification purpose, the i-th multivariate time trajectory is formalized by the matrix $\mathbf{X}_i \equiv \{\mathbf{x}_{it}: \ t = 1, \ldots, T\}$, where $\mathbf{x}_{it} \equiv (x_{i1t}, \ldots, x_{ijt}, \ldots, x_{iJt}), \ i = 1, \ldots, I, \ t = 1, \ldots, T$.

2.1. Dynamic Time Warping

The Dynamic Time Warping (DTW) (Velichko et al., 1970; Berndt, 1994; Izakian et al., 2015; D'Urso et al., 2018) allows to locally stretch or compress multivariate time series to make their shape as similar as possible.

To this end, the functions that allow to remap each multivariate time series need to be identified. This kind of function is called warping function and its aim is to "realign" the time indices of the multivariate time series.

Given a "query" (or test) multivariate time series $\mathbf{X}_i$ and a "reference" multivariate time series, $\mathbf{X}_{i'}$, with length T and T' ($T \gtreqless T'$) respectively, the total distance between $\mathbf{X}_i$ and $\mathbf{X}_{i'}$ is computed by means of the warping path. The warping path allows to compare each data point in $\mathbf{X}_i$ with the closest data point in $\mathbf{X}_{i'}$, and is defined as

$$\Phi_l = (\varphi_l, \psi_l), \ l = 1, \ldots, L.$$

under the following constraints.

1. boundary condition: $\Phi_1 = (1, 1), \ \Phi_L = (T, T')$;
2. monotonicity condition: $\varphi_1 \leq \ldots \leq \varphi_l \leq \ldots \leq \varphi_L$ and $\psi_1 \leq \ldots \leq \psi_l \leq \ldots \leq \psi_L$.

The total dissimilarity between the two "warped" multivariate time series is:

$$\sum_{l=1}^{L} d(\mathbf{x}_{i,\varphi_l}, \mathbf{x}_{i',\psi_l}) m_{l,\Phi}$$

where $m_{l,\Phi}$ is a local weighting coefficient, and $d(.,.)$ is, usually, the Euclidean distance for multivariate time series (Giorgino et al., 2009). Since there are several warping curves, the DTW dissimilarity measure is the one which correspond to the optimal warping curve, $\hat{\Phi}_l = (\hat{\varphi}_l, \hat{\psi}_l), \ (l = 1, \ldots, L)$, which minimizes the total dissimilarity between $\mathbf{X}_i$ and $\mathbf{X}_{i'}$:

$$D(\mathbf{X}_i, \mathbf{X}_{i'}) = \min_{\Phi_l} \sum_{l=1}^{L} d(\mathbf{x}_{i,\varphi_l}, \mathbf{x}_{i',\psi_l}) m_{l,\Phi} = \sum_{l=1}^{L} d(\mathbf{x}_{i,\hat{\varphi}_l}, \mathbf{x}_{i',\hat{\psi}_l}) m_{l,\hat{\Phi}}. \tag{2}$$

The DTW dissimilarity measure is particularly useful when comparing multivariate time series. First, by preserving the time ordering of the sequence, the DTW goes beyond the instantaneous features of time data. Indeed, DTW dissimilarity measure copes with both the instantaneous and the variational features of the multivariate time trajectories, i.e., the instantaneous position of the trajectories and their dynamic evolution over time, thus providing a more complete comparison that takes into account also the different rates at which phenomena change over times. Second the DTW dissimilarity measure is also more flexible than the Euclidean distance since it allows for comparison of multivariate time series of different lengths. Third, no assumptions are required regarding the multivariate time series properties. Furthermore, Euclidean distance is calculated in a one-to-one manner, while DTW dissimilarity measure tries to find the best warping. Finally, by taking explicitly into account the ordering of the observations, DTW also deals with the presence of possible time shits in the data.

For all these reasons, DTW is now usually adopted as a suitable alternative to Euclidean distance in time series cluster analysis (see, among others, Berndt, 1994; Oates et al., 1999; Jeong et al., 2011; Petitjean et al., 2011; Begum et al., 2015; Izakian et al., 2015; Mure et al., 2016) In particular, Ding et al. (2008) and Rakthanmanon et al. (2012) experimentally proved the effectiveness of DTW in data mining problems—like time series clustering is—with respect to other distance measures.

Furthermore, while DTW is more computationally demanding than Euclidean distance, by adopting a Partitioning-Around-Medoids (PAM, Kaufman & Rousseeuw, 2005) approach (see section 2.3 below), the distance matrix should be computed only once at the start of the overall clustering procedure (D'Urso et al., 2018).

2.2. Dealing with space: proximity matrix

When dealing with spatial data the within group dispersion has to be minimised and the spatial autocorrelation between contiguous spatial units has to be taken into consideration. This spatial information can be analytically embedded in the clustering process using a "proximity" matrix, say $\mathbf{P}$, that is a symmetric matrix of order I whose elements signal the proximity between two spatial areas (Pham, 2001; Coppi et al., 2010). In the literature, there are different ways of defining proximity and consequently there are different ways of constructing proximity matrices among spatial units (Gordon, 1999; Páez & Scott, 2005). Two of the most common definitions are based on connectivity, i.e. travel time or distance between pairs of units, and physical contiguity.

Connectivity can be coped with by means of a proximity matrix $\mathbf{P}$ whose elements are given by the inverse of a generic measure of the distance between i and i' (distance between the two spatial units, trip duration and/or cost, etc.), normalized to range in $[0, 1]$. The more two spatial areas are connected, the lower is the value in the proximity matrix. Obviously, diagonal elements are all equal to 0.

Spatial contiguity, on its turn, can be specified in several ways. For instance, two spatial units can be contiguous either if they are adjacent (neighbours) or if they belong to the same macro-area, even if they are not adjacent. In this case, $\mathbf{P}$ is constructed as a symmetric matrix with 0 diagonal elements and with off-diagonal elements given by:

$$p_{ii'} = \begin{cases} 1 & \text{if } i \text{ is contiguous to } i' \\ 0 & \text{otherwise} \end{cases} \qquad i = 1, \dots, I, \ i \neq i'. \tag{3}$$

2.3. The DTW-Fuzzy C-Medoids clustering algorithm for Spatial-Temporal Trajectories (DTW-FCMd-STT)

In this paper, following a PAM apprach in a fuzzy framework, the Fuzzy C-Medoids (FCMd, Krishnapuram et al., 2001) clustering algorithm is adopted. With respect to standard (crisp) clustering algorithms, fuzzy clustering algorithms are generally more efficient—dramatic changes in the value of cluster membership are less likely to occur in estimation procedures—and they are less affected by both local optima and convergence problems (Everitt et al., 2001; Hwang et al., 2007). With complex data as multivariate time series are, it could be difficult to identify a clear boundary between clusters in real applications. In this sense, fuzzy clustering appears more attractive than the crisp clustering methods ?Wedel & Kamakura (2000). Finally, the membership degrees produced by fuzzy clustering methods, that indicate the belonging of each unit to each cluster, also indicate whether there is a second-best cluster almost as good as the best cluster, a scenario which crisp clustering methods cannot uncover Everitt et al. (2001).

Regarding the choice of the fuzzy clustering method, with respect to Fuzzy C-Means (FCM, Bezdek, 1981), FCMd allows for more appealing and easy to interpret results of the final partition (Kaufman & Rousseeuw, 2005) by obtaining non-fictitious representative time series (i.e. the medoids) as final result (see section 2.6).

Dealing with Spatial-Temporal trajectories, possible spillover effects between adjacent units have to be taken into account. As observed in section 2.2, since there could be different, say $K\,(K \geq 1)$, definitions of proximity, K spatial penalty terms are added to the objective function. Following Pham & Prince (1999) and Coppi et al. (2010), the DTW-Fuzzy C-Medoids clustering algorithm for Spatial-Temporal Trajectories (DTW-FCMd-STT) is then formalised as follows:

$$\begin{cases} \min: & \sum\limits_{i=1}^{I}\sum\limits_{c=1}^{C} u_{ic}^{m} D(\mathbf{X}_i, \widetilde{\mathbf{X}}_c) + \sum\limits_{k=1}^{K} \frac{\beta_k}{2} \sum\limits_{i=1}^{I}\sum\limits_{c=1}^{C} u_{ic}^{m} \sum\limits_{i'=1}^{I}\sum\limits_{c' \in C_c} p_{kii'} u_{i'c'}^{m} \\ s.t. & \sum\limits_{c=1}^{C} u_{ic} = 1,\ u_{ic} \geq 0 \end{cases} \tag{4}$$

where $\mathbf{X}_i$ and $\widetilde{\mathbf{X}}_c$ are the multivariate time trajectories of the i-th spatial unit and of the c-th spatial medoid ($c = 1, \ldots, C$), respectively; $D(\cdot, \cdot)$ is the DTW dissimilarity measure for multivariate spatial time series; $m > 1$ is the fuzziness parameter; $\beta_k \geq 0$ is the tuning parameter of the k-th spatial information; $p_{kii'}$ is the generic element of the $(I \times I)$ "proximity" matrix $\mathbf{P}_k$; C_c is the set of the C clusters, with the exclusion of cluster c; u_{ic} is the membership degree of the unit i to the cluster c.

The objective function in (4) is made up by two distinguished terms:

- the time dependent term (see section 2.3.1)

$$\sum_{i=1}^{I}\sum_{c=1}^{C} u_{ic}^{m} D(\mathbf{X}_i, \widetilde{\mathbf{X}}_c); \tag{5}$$

- the spatial dependent term

$$\sum_{k=1}^{K} \frac{\beta_k}{2} \sum_{i=1}^{I}\sum_{c=1}^{C} u_{ic}^{m} \sum_{i'=1}^{I}\sum_{c' \in C_c} p_{kii'} u_{i'c'}^{m} \tag{6}$$

  which is the sum of K spatial penalty terms (see section 2.3.2).

The two terms (5) and (6) are computed over the same data range, i.e., over the same observations. In the clustering process, one term could dominate the other depending on the data at hand. The way in which both terms contribute to the clustering results will be clarified in sections 2.3.1-2.3.2.

The optimal iterative solution for the objective function in (4) is:

$$u_{ic} = \frac{\left[D(\mathbf{X}_i, \widetilde{\mathbf{X}}_c) + \sum_{k=1}^{K} \beta_k \sum_{i'=1}^{I} \sum_{c' \in C_c} p_{kii'} u_{i'c'}^m\right]^{-\frac{1}{m-1}}}{\sum_{c'=1}^{C} \left[D(\mathbf{X}_i, \widetilde{\mathbf{X}}_{c'}) + \sum_{k=1}^{K} \beta_k \sum_{i'=1}^{I} \sum_{c'' \in C_{c'}} p_{kii'} u_{i'c''}^m\right]^{-\frac{1}{m-1}}} \tag{7}$$

As a final remark, the overall optimization of the objective function in (4) ensures that the cohesion within clusters is maximized and that the spatial autocorrelation existing in the data at hand is properly coped with, simultaneously, as it will be explained in the following.

2.3.1. Time dependent term

The time dependent term (5) is the within cluster dispersion due to the time-varying features of multivariate trajectories. As observed in section 2.1, in this term the whole time information is inherited by the Dynamic Time Warping measure, that takes into account both the instantaneous and the variational features of the multivariate time trajectories. When there are no spatial information, the time dependent term (5) coincides with the Dynamic Time Warping Fuzzy C-Medoids (DTW-FCMd) for multivariate time trajectories introduced by D'Urso et al. (2018).

2.3.2. Spatial dependent term

The spatial dependent term (6) suitably allows the objective function to incorporate different sources of spatial information. The term (6) is the sum of $K\,(K \geq 1)$ spatial penalty terms (Pham, 2001; Coppi et al., 2010), one for each definition of proximity among areas considered. In this way, the clustering method captures the information connected to the different levels of proximity (multilevel proximity). For instance, we can consider the simple case illustrated in Figure 2 in which 5 units, i.e. towns, and 2 macroarea, i.e. valleys, are considered. In this specific case, two kinds of proximity can be defined: (i) proximity among towns (level 1 proximity); belonging to the same valley (level 2 proximity). Therefore, four different scenarios can be identified: 1) two towns (a_1 and a_2) are close to each other (level 1 proximity) and they belong to the same valley (level 2 proximity); 2) two towns (a_1 and b_1) are close to each other (level 1 proximity) but they do not belong to the same valley; 3) two towns (a_1 and a_3) are not close to each other but they belong to the same valley (level 2 proximity); 4) two towns (a_1 and b_2) are not close to each other and they do not belong to the same valley.

In each spatial penalty term two parameters are relevant, the proximity matrix $\mathbf{P}_k$, and the tuning parameter β_k.

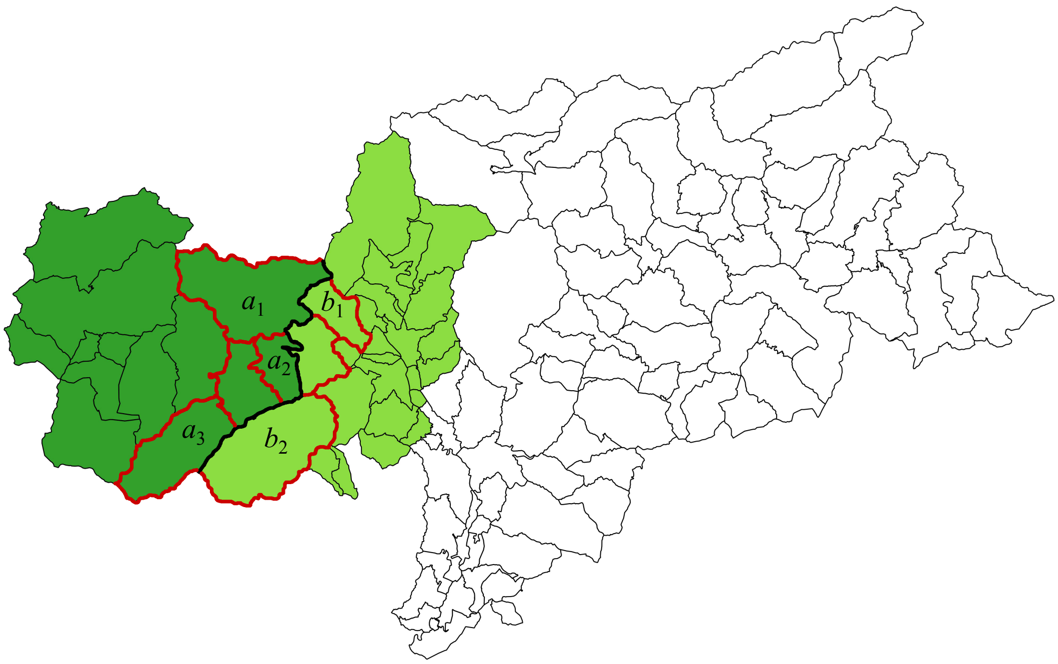


Figure 2: Example of proximity among areas where a_1, a_2, a_3, b_1, and b_2 are towns and the light green and dark green areas represent two valleys

The role of the k-th proximity matrix, $\mathbf{P}_k$, is to increase the membership degree of unit i in cluster c and, at the same time, to increase the membership degrees of the units that are connected, in some way, to i in cluster c, while reducing these membership degrees in the other clusters. We define this spatial smoothing as "proximity effect", where, as previously observed, the concept of proximity is vast enough to encompass different types of connectivity between areas. The tuning parameter β_k must be set depending on the spatial autocorrelation among data (see section 2.5 below). β_k could enhance the proximity effect due to $\mathbf{P}_k$ if the spatial autocorrelation between units is high, e.g., if the features of a spatial unit display a certain degree of concordance with those of its neighbours. Otherwise, β_k could counterbalance, if not neutralise at all, the proximity effect, if there is relatively low spatial autocorrelation between areas. Then, the greater the value of β_k, the greater is the weight of the concept of proximity ascribed to it in the clustering process. Let say that β_1 corresponds to the distance between areas, and β_2 to the belonging to the same macro-area, then, if $\beta_1 > \beta_2$, "closeness" plays a major role than "belonging" in the optimization process.

As already observed, the choice of the value of β_k is data dependent. Coppi et al. (2010) observed that the choice should be made according to a measure of a within cluster spatial autocorrelation (see section 2.5), to avoid that the spatial smoothing induced by the proximity matrix overcome the cluster separation. Indeed, an excessively high value of one or more β_k's could constraint all neighbour units to be classified in one cluster, regardless the features observed. A heuristic procedure for a custom-made choice of β_k's is illustrated in section 4.

Finally, it should be stressed that by combining $\mathbf{P}_k$ and β_k in the clustering process, we are able to take into account also the spatial autocorrelation which is more informative than the spatial proximity alone.

2.3.3. A remark on the use of spatial information

As highlighted in the Introduction, in spatial clustering there are different approaches to incorporate spatial information in a clustering framework. In particular, spatial information can be represented in a clustering method by considering the contiguity/adjacencies between each pair of territorial (spatial) units (Gordon, 1999). This information is usually formalized in the clustering method by means of contiguity/adjacencies constraints or suitable spatial weights associated to distance measures. This approach is preferred in hierarchical clustering (i.e. agglomerative) or in relational clustering where the distance measure is taken for each pair of territorial units. In doing so the spatial information is represented algebraically by a squared matrix (called either contiguity matrix or spatial matrix) associated to the squared distance matrix. Each element of this matrix represents the territorial proximity between two units that can be represented by either dichotomous values (0 or 1), indicating if the units are neighbouring or not, or quantitative values representing the road distances or travel times.

In the literature, another well-known approach used to incorporate spatial information in the clustering procedure is to introduce a suitable penalty term in the objective function used in the optimization procedure for clustering territorial units (see, e.g., Pham, 2001; **?**; Coppi et al., 2010). This approach is used in non-hierarchical framework (e.g. hard or fuzzy C-means clustering and hard or fuzzy partitioning around medoids procedures, as the hard or fuzzy C-medoids clustering), where the spatial information cannot be represented by squared matrix. In fact, in these cases, the dimension of the distance matrix is rectangular (the matrix contains values representing, e.g., the distance between each territorial unit and each centroid or between each territorial unit and each medoid, where centroids and medoids are the prototypes representing the clusters). This approach is quite common in the spatial clustering literature. As remarked by Pham (2001), "a classical approach to incorporating spatial information is to penalize the [...] objective function [of the fuzzy clustering] to constrain the behavior of the membership functions, similar to methods used in regularization and Markov random field (MRF) theory (**?**). This penalty can be used to discourage unlikely or undesirable configurations in the membership functions, such as a high membership value immediately surrounded by low values of the same class". The Markov random field (MRF) theory has been used by **?** "which used standard first order differences as a penalty to force membership values to be similar to neighbouring values. The main problem with such a penalty function, however, is that it can drastically alter the characteristics of the membership function in an undesirable fashion. For example, first order differences will cause membership functions to be nearly piecewise constant. Second order differences will cause membership functions to be more smooth. However, depending on the value of the [m] parameter, this may contradict the desired characteristics of the membership functions. [In our method], the objective function [see formula (4)] includes a penalty term that is reminiscent of MRF priors but is consistent with the desired behavior of the membership functions dictated by the value of the m parameter" (Pham, 2001). As remarked before, the use of penalty terms for taking into account the spatial proximity is largely used in the literature, in different research areas (see, among others, **??????????**). Then, it is a consolidated methodological approach in the spatial clustering analysis.

Notice that, since our clustering method classify territorial units following a non-hierarchical approach, we cannot consider the spatial information represented by contiguity or spatial measures (that compare pair of units) formalized as constraints or weights associated to distance matrix (as in the hierarchical approach). In addition, since we consider different levels of contiguity, considering different adjacency matrices as weights to embed would considerably increase the complexity of the procedure. Nonetheless, as will be remarked in the Conclusions, in the future we will explore the possibility to take into account the spatial information in the clustering process following another clustering approach, i.e. the fuzzy relational method (**?**Kaufman & Rousseeuw, 2005; D'Urso, 2015). We will aslo investigate the computational and operational complexity of this alternative clustering procedure (scalability, etc.).

2.4. Validity measure

In general, internal validity measures provide useful guidelines in the identification of the best partition (Handl et al., 2005; D'Urso, 2015). Suitable measures for fuzzy clustering algorithm have been suggested by Xie & Beni (1991) and Campello & Hruschka (2006).

The Xie and Beni cluster validity index (Xie & Beni, 1991) is the ratio between compactness and separation among clusters and it can be expressed as:

$$XB = \frac{\sum\limits_{i=1}^{I}\sum\limits_{c=1}^{C} u_{ic}^{p} D(\mathbf{X}_i, \widetilde{\mathbf{X}}_c)}{I \min\limits_{p \neq q} D(\widetilde{\mathbf{X}}_p, \widetilde{\mathbf{X}}_q)} \tag{8}$$

where $(p, q) \in \{1 \ldots, C\}$. The smaller XB, the more compact and separate are the clusters.

The Fuzzy Silhouette (FS) index (Campello & Hruschka, 2006) is computed as the weighted average of individual silhouettes width, λ_i, (Kaufman & Rousseeuw, 2005), with weights derived from the fuzzy membership matrix $\mathbf{U} = \{u_{ic} : \ i = 1, \ldots, I; \ c = 1, \ldots, C\}$ as follows:

$$FS = \frac{\sum_{i=1}^{I}(u_{ip} - u_{iq})^{\alpha} \cdot \lambda_i}{\sum_{i=1}^{I}(u_{ip} - u_{iq})^{\alpha}}, \qquad \lambda_i = \frac{(b_i - a_i)}{\max\{b_i, a_i\}} \tag{9}$$

Here, a_i is the average distance between the i-th unit and the units belonging to the cluster p ($p = 1$,...,C) with which i is associated with the highest membership degree; b_i is the minimum (over clusters) average distance of the i-th unit to all units belonging to the cluster q with $q \neq p$; $(u_{ip} - u_{iq})^{\alpha}$ is the weight of each λ_i calculated upon $\mathbf{U}$, where p and q are, respectively, the first and second best clusters (accordingly to the membership degree) to which the i-th unit is associated; $\alpha \geq 0$ is an optional user defined weighting coefficient. The traditional (crisp) Silhouette coefficients is obtained by setting $\alpha = 0$. The higher the value of FS, the better the assignment of the units to the clusters simultaneously obtaining the minimisation of the intra-cluster distance and the maximisation of the inter-cluster distance.

2.5. Spatial autocorrelation

In this paper, we introduce a new measure of spatial autocorrelation to assess the post-cluster autocorrelation between units, the Fuzzy Moran (FM) index. This index is a multivariate fuzzy generalisation of the Moran's index (Gittleman & Kot, 1990) and it is a generalization of the spatial autocorrelation measure introduced by Coppi et al. (2010). The idea of the FM index is to compute the spatial autocorrelation between classified units in which both the fuzzy membership matrix $\mathbf{U}$ and the spatial proximity matrices $\mathbf{P}_k$ are considered. The FM index is defined as follows:

$$FM = \frac{tr\left[\bar{\mathbf{X}}'\mathbf{U}_c^{\frac{1}{2}}\tilde{\mathbf{P}}\mathbf{U}_c^{\frac{1}{2}}\bar{\mathbf{X}}\right]}{tr\left[\bar{\mathbf{X}}'\mathbf{U}_c^{\frac{1}{2}}diag(\tilde{\mathbf{P}}'\tilde{\mathbf{P}})\mathbf{U}_c^{\frac{1}{2}}\bar{\mathbf{X}}\right]} \tag{10}$$

where $\mathbf{U}_c$ is the square diagonal matrix of order I of the membership degrees of cluster c; $\bar{\mathbf{X}}$ is the centred "compromise" matrix (mean of the T data matrices $\mathbf{X}_t$); $\tilde{\mathbf{P}}$ is the weighted spatial matrix obtained as linear combination between the K proximity matrices as follows

$$\tilde{\mathbf{P}} = \sum_{k=1}^{K} w_k \mathbf{P}_k \tag{11}$$

where $0 \leq w_k \leq 1$ and $\sum_{k=1}^{K} w_k = 1$. The FM index (as the Moran's index) ranges between -1 and 1. A value of 1 indicates perfect positive spatial autocorrelation, i.e. neighbouring units have similar values, 0 indicates no autocorrelation, i.e. units are spatially random located, and -1 indicates perfect negative spatial autocorrelation, i.e. neighbouring units have dissimilar values (Páez & Scott, 2005). Thus, the higher the FM value, the better the geographical assignment of the units to the clusters.

Moreover, the Fuzzy Moran's index, as the Moran's index, can be interpreted as a measure of spatial spill-over effect (Ma et al., 2015; Yang, 2012). In the literature, the spatial spill-over effect is considered as the indirect or unintentional effects that a geographical area exerts on other neighbour areas (Yang & Fik, 2014). A positive spill-over effect is obtained when an area benefits of their neighbours influence due to the existence of spatial externalities across area.

2.6. Some comparative assessment

Our proposal inherits all the advantages of the ingredients considered in the methodological framework. In particular, in a comparative assessment point of view, with respect to some methods suggested in the literature we have the following evidences.

- The fuzzy clustering methods proposed by D'Urso et al. (2018) show very good performance for clustering units with time-varying information. However, when the units are regions, geographical areas, etc., it is more useful to analyse this kind of units by considering clustering methods capable to capture the territorial nature of the units. To this purpose, the method proposed in this paper is able to cluster units not only considering time information but also taking into account additional information connected to spatial characteristics of the units. In particular, our method is able to cluster territorial units considering explicitly in the objective function the spatial information connected to the units—territorial proximity and spatial autocorrelation (see sections 2.2 and 2.3.2). Notice that, the fuzzy clustering methods proposed by D'Urso et al. (2018) could be applied to territorial units, but ignoring the territorial information that characterizes this type of unit. However, this would represent a relevant loss of information in the spatial analysis process. Furthermore, with respect to the fuzzy clustering methods suggested by D'Urso et al. (2018) based on the Euclidean distance, the proposed method inherits all the advantages of the DTW-based dissimilarity measure (see, section 2.1).

- The Fuzzy C-Means clustering method for spatial time series proposed by Coppi et al. (2010) (Cross-Sectional Fuzzy C-Means for Spatial-Temporal Trajectories, CS-FCM-STT) is able to cluster territorial units with time-varying information. With respect to this method our proposal has two more advantages inherited: (i) by the kind of prototypes utilized in our method (i.e. medoids vs centroids); (ii) by the characteristics of the spatial component considered in the objective function of the proposed method.

  (i) With respect to the advantage connected to the kind of prototypes (i.e. medoids), adopting PAM approach, the prototypes of each cluster (medoids) are territorial units actually observed and not "virtual" territorial units like the "centroids" derived with a Fuzzy C-Means—as in the method suggested by Coppi et al. (2010). Overall, having non-fictitious representative territorial units available makes interpreting the obtained clusters easier, which is often very useful in geographical and territorial applications. In fact, "in many clustering problems one is particularly interested in a characterization of the clusters by means of typical or representative objects [territorial units]. These are objects [territorial units] that represent the various structural aspects of the set of objects [territorial units] being investigated. There can be many reasons for searching for representative objects [territorial units]. Not only can these objects [territorial units] provide a characterization of the clusters, but they can often be used for further work or research, especially when it is more economical or convenient to use a small set of k objects [C territorial units in our case] instead of the large set one started off with" (Kaufman & Rousseeuw, 2005). Furthermore, PAM clustering approach is slightly more robust than C-Means approach (Garcia-Escudero & Gordaliza, 1999; **?**; Estivill-Castro & Yang, 2004; Kaufman & Rousseeuw, 2005), hence DTW-FCMd-STT is relatively more resistant to the presence of noise in the data than CS-FCM-STT.

(ii) With respect to the advantages connected to spatial dependent term of the objective function, our spatial term is more general compared with the spatial term considered in the method suggested by Coppi et al. (2010). In fact, as remarked in section 2.3.2, it is capable to consider different level of spatial proximity (multilevel proximity) and then it is more informative in a spatial point of view in the sense that it is able to capture in deep the political and physical geographical characteristics -e.g. administrative and economic features and geophysical and orographic nuances- of the analysed territorial area. In this way, the spatial dependent term used in Coppi et al. (2010) is a particular case of the term adopted in our method. See section 2.3.2 for more details.

3. Illustration with simulated data

3.1. Simulation study 1

In the following, a simulation study in which two contiguity matrices are considered for simplicity, is presented. The aim of this exercise is to assess the sensitivity of the clustering process to the contiguity matrices, according to the k-th spatial parameters β_k (formula 4).

An artificial data set is generated with two natural clusters and two units close to each other and characterized by soft memberships to one of the two clusters. Two contiguity matrices, one with contiguity only among the units within the natural clusters (including the soft membership unit) and one including the contiguity between the soft membership units as well, are generated. The aim of the simulation is to verify the decreasing of the fuzzy membership degrees of the two soft membership units with respect to their natural clusters and, eventually, even their memberships to the same cluster while increasing the spatial penalty coefficient of the matrix including contiguity between them. For this reason, the spatial penalty coefficients β_1 and β_2 range in (0, 8).

The number of units, variables, and periods of time considered are $I = 8$, $J = 2$, and $T = 6$, respectively. In the contiguity matrix $\mathbf{P}_2$, two sets of contiguous units are defined, i.e. (1, 2, 3, 4) and (5, 6, 7, 8), whereas in $\mathbf{P}_1$ the contiguity between units 4 and 5 is added. The contiguity matrices $\mathbf{P}_1$ and $\mathbf{P}_2$ are the following:

$$\mathbf{P}_1 = \begin{array}{cc} & \begin{array}{cccccccc} u1 & u2 & u3 & u4 & u5 & u6 & u7 & u8 \end{array} \\ \begin{array}{c} u1 \\ u2 \\ u3 \\ u4 \\ u5 \\ u6 \\ u7 \\ u8 \end{array} & \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix} \end{array}$$

$$
\mathbf{P}_2 = \begin{array}{c} \\ u1 \\ u2 \\ u3 \\ u4 \\ u5 \\ u6 \\ u7 \\ u8 \end{array}
\left(\begin{array}{cccccccc}
u1 & u2 & u3 & u4 & u5 & u6 & u7 & u8 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 1
\end{array}\right)
$$

The generation process of the dataset is summarized in Table 1. The defined clusters are (1, 2, 3) and (6, 7, 8). Units 4 and 5 are characterized by a *fuzzy* membership to clusters (1, 2, 3) and (6, 7, 8), respectively. Going from data configuration 1) to data configuration 4), we can note that units 4 and 5 are getting closer and closer (Table 1 and Figure 3).

Configuration		**units 1,2,3**	**unit 4**	**unit 5**	**units 6,7,8**
1)	j=1	$U[0.0, 0.5]$	$U[0.8, 1.0]$	$U[1.0, 1.2]$	$U[1.5, 2.0]$
	j=2	$U[0.0, 0.5]$	$U[0.8, 1.0]$	$U[1.0, 1.2]$	$U[1.5, 2.0]$
2)	j=1	$U[0.0, 0.5]$	$U[0.85, 1.0]$	$U[1.0, 1.15]$	$U[1.5, 2.0]$
	j=2	$U[0.0, 0.5]$	$U[0.85, 1.0]$	$U[1.0, 1.15]$	$U[1.5, 2.0]$
3)	j=1	$U[0.0, 0.5]$	$U[0.9, 1.0]$	$U[1.0, 1.1]$	$U[1.5, 2.0]$
	j=2	$U[0.0, 0.5]$	$U[0.9, 1.0]$	$U[1.0, 1.1]$	$U[1.5, 2.0]$
4)	j=1	$U[0.0, 0.5]$	$U[0.95, 1.0]$	$U[1.0, 1.05]$	$U[1.5, 2.0]$
	j=2	$U[0.0, 0.5]$	$U[0.95, 1.0]$	$U[1.0, 1.05]$	$U[1.5, 2.0]$

Table 1: Data generation process for simulation study 1. Two clusters are generated from the data. Going from configuration 1) to configuration 4), units 4 and 5 are getting closer

The membership degree obtained in the case of the fourth data configuration (see Table 1 and Figure 3) are reported in Table 2. By suitably tuning the values of β_1 and β_2, and therefore the separate influence of the two contiguity matrices $\mathbf{P}_1$ and $\mathbf{P}_2$, we can see how the two units 4 and 5 become more clearly separated, and then classified to the respective clusters when $\beta_1 < \beta_2$, or, on the contrary, are classified in the same cluster, when $\beta_1 > \beta_2$.

For more details on the membership degrees and on performance results, see the Appendix Appendix A.1 to this paper.

3.2. Simulation study 2

This simulation study is similar to that presented in section 3.1. We increased the number of objects and of clusters, to show the performance of DTW-FCMd-STT in a more complex environment. Similarly as in an simulation study 1, artificial data set is generated with four natural clusters and four units close to each other characterized by soft membership to one of the four clusters. Two contiguity matrices, one with contiguity

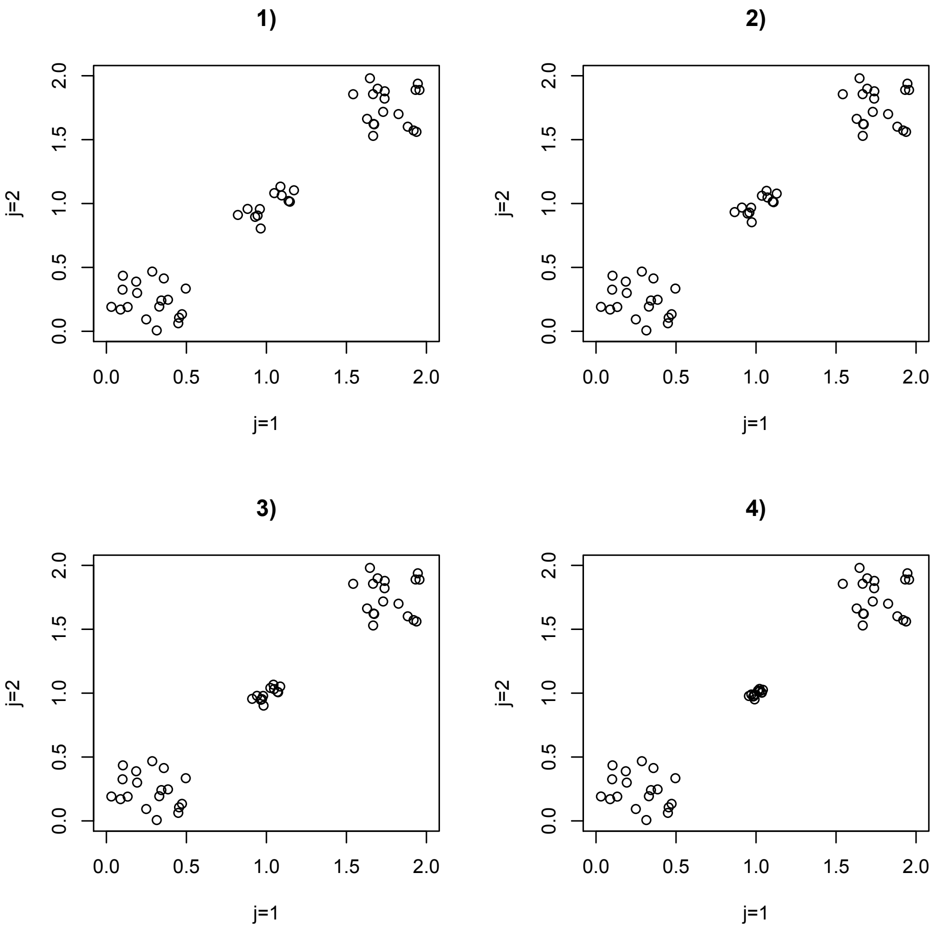


Figure 3: Data generation process for simulation study 1. Two clusters are generated from the data. Going from configuration 1) to configuration 4), units 4 and 5 are getting closer

(β_1, β_2)	(0, 0)		(4, 0)		(0, 4)		(8, 0)		(0, 8)	
cluster	1	2	1	2	1	2	1	2	1	2
1	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
2	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
3	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**
4	0.6271	0.3729	0.6853	0.3147	0.6689	0.3311	0.7168	0.2832	0.7051	0.2949
5	0.4874	0.5126	0.5256	0.4744	0.4603	0.5397	0.5169	0.4831	0.4138	0.5862
6	0.0000	1.0000	0.0000	1.0000	0.0001	0.9999	0.0000	1.0000	0.0001	0.9999
7	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**
8	0.0003	0.9997	0.0001	0.9999	0.0004	0.9996	0.0001	0.9999	0.0004	0.9996

Note: Medoids' membership degrees are in bold.

Table 2: Membership degrees for simulation study 1 obtained under the data configuration 4), according to different combinations of β_1 and β_2

only among the units within the natural clusters (including the soft membership unit) and one including the contiguity among the soft membership units as well, are generated. The aim of the simulation is to verify the decreasing of the fuzzy membership degree of the four soft membership units to the natural clusters and eventually even their membership to the same cluster while increasing the spatial penalty coefficient of the matrix including contiguity among them. To this end, the spatial penalty coefficients β_1 and β_2 range in (0, 20).

The number of units, variables, and periods of time considered are $I = 16$, $J = 2$, and $T = 6$, respectively. In the first contiguity matrix ($\mathbf{P}_2$), the contiguous units are (1, 2, 3, 4), (5, 6, 7, 8), (9, 10, 11, 12) and (13, 14, 15, 16), whereas in $\mathbf{P}_1$ the contiguity among units 4, 5, 12, 13 is added.

The generation process of the dataset is summarized in Table 3. The defined clusters are (1, 2, 3), (6, 7, 8), (9, 10, 11), (14, 15, 16). Units 4, 5, 12, 13 are characterized by a *fuzzy* membership to clusters (1, 2, 3), (6, 7, 8), (9, 10, 11), (14, 15, 16), respectively. Going from data configuration 1) to data configuration 4) units 4, 5, and 12, 13 are getting closer and closer, respectively (Table 3 and Figure 4).

Configuration		**units 1,2,3**	**unit 4**	**unit 5**	**units 6,7,8**	**units 9,10,11**	**unit 12**	**unit 13**	**units 14,15,16**
1)	j=1	$U[0.0, 0.5]$	$U[0.6, 0.7]$	$U[0.6, 0.7]$	$U[0.0, 0.5]$	$U[1.5, 2.0]$	$U[1.3, 1.4]$	$U[1.3, 1.4]$	$U[1.5, 2.0]$
	j=2	$U[0.0, 0.5]$	$U[0.6, 0.7]$	$U[1.3, 1.4]$	$U[1.5, 2.0]$	$U[1.5, 2.0]$	$U[1.3, 1.4]$	$U[0.6, 0.7]$	$U[0.0, 0.5]$
2)	j=1	$U[0.0, 0.5]$	$U[0.7, 0.8]$	$U[0.7, 0.8]$	$U[0.0, 0.5]$	$U[1.5, 2.0]$	$U[1.2, 1.3]$	$U[1.2, 1.3]$	$U[1.5, 2.0]$
	j=2	$U[0.0, 0.5]$	$U[0.7, 0.8]$	$U[1.2, 1.3]$	$U[1.5, 2.0]$	$U[1.5, 2.0]$	$U[1.2, 1.3]$	$U[0.7, 0.8]$	$U[0.0, 0.5]$
3)	j=1	$U[0.0, 0.5]$	$U[0.8, 0.9]$	$U[0.8, 0.9]$	$U[0.0, 0.5]$	$U[1.5, 2.0]$	$U[1.1, 1.2]$	$U[1.1, 1.2]$	$U[1.5, 2.0]$
	j=1	$U[0.0, 0.5]$	$U[0.8, 0.9]$	$U[1.1, 1.2]$	$U[1.5, 2.0]$	$U[1.5, 2.0]$	$U[1.1, 1.2]$	$U[0.8, 0.9]$	$U[0.0, 0.5]$
4)	j=1	$U[0.0, 0.5]$	$U[0.9, 1.0]$	$U[0.9, 1.0]$	$U[0.0, 0.5]$	$U[1.5, 2.0]$	$U[1.0, 1.1]$	$U[1.0, 1.1]$	$U[1.5, 2.0]$
	j=2	$U[0.0, 0.5]$	$U[0.9, 1.0]$	$U[1.0, 1.1]$	$U[1.5, 2.0]$	$U[1.5, 2.0]$	$U[1.0, 1.1]$	$U[0.9, 1.0]$	$U[0.0, 0.5]$

Table 3: Data generation process for simulation study 2. Four clusters are generated from the data. Going from configuration 1) to configuration 4), units 4, 5, and 12, 13 are getting closer

Once again, according to the combination of β_1 and β_2, the *fuzzy* units get more separated when $\beta_1 < \beta_2$, while eventually are classified in the same cluster when $\beta_1 > \beta_2$.

For more details on the membership degrees and on performance results, see the Appendix Appendix A.2 to this paper.

3.3. Simulation study 3

In this simulation study we highlight two main features of the proposed clustering method:

1. the capability to deal with time series of different length;
2. the capability to fully exploit spatial information.

We simulated a dataset of 20 three-variate ($I = 20$, $J = 3$) time series of length ranging from $T = 6$ to $T = 10$. The data generation process yielded to three partially overlapping clusters ($C = 3$) of size 10, 5 and 5, respectively (see Figure 5).

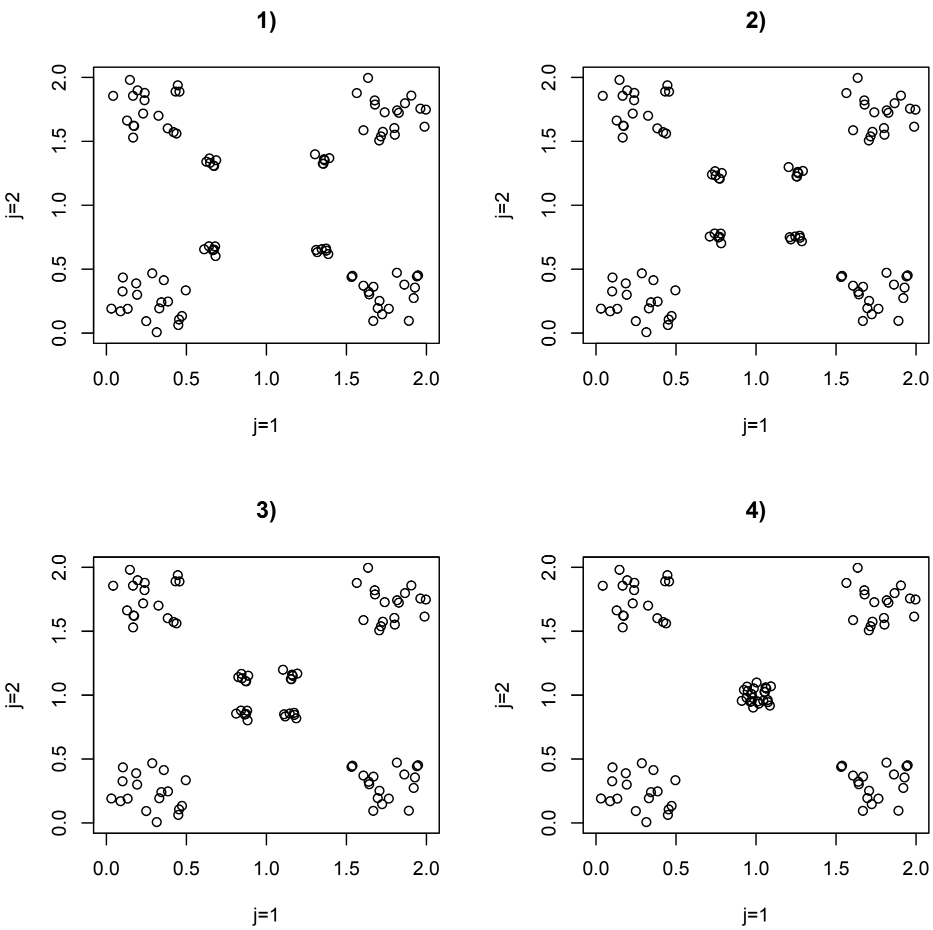


Figure 4: Data generation process for simulation study 2. Four clusters are generated from the data. Going from configuration 1) to configuration 4), units 4, 5, and 12, 13 are getting closer

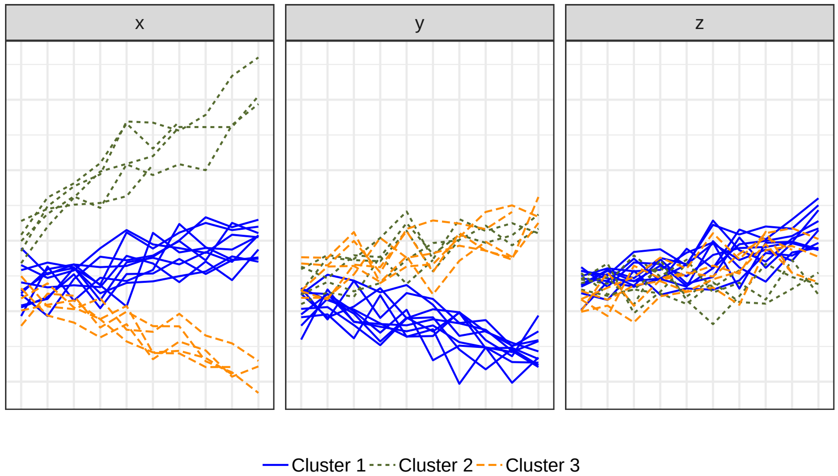


Figure 5: Simulated data for simulation study 3. Data are generated to be classified into three partially overlapping clusters. Time series belonging to different clusters are depicted with different colours and line types

As for the spatial dependence, we generated two proximity matrices, $\mathbf{P}_1$ and $\mathbf{P}_2$, illustrated in Figure 6. A black square indicate that there is some kind of proximity between i and j. The two matrices refer to different notions of proximity, which are closely related to those observed in the empirical application: $\mathbf{P}_1$ refers to a situation in which two units are neighbours if they share a border; $\mathbf{P}_2$ represents a situation in which proximity is due to the fact that belong to the same macro-area, even if they are not neighbour. Furthermore, each macro-area corresponds to a different cluster. By observing $\mathbf{P}_1$ and $\mathbf{P}_2$, there are some units that are neighbours even if they belong to different macro-areas, and some units that belong to the same macro-area but they are not neighbour. Finally, the parameters β_1 and β_2 are set to 0 or 1.8, depending on how the spatial information is exploited.

The purpose of the present simulation is to show the capability of DTW-FCMd-TSS to individuate the three clusters, even if data are rather noise, by exploiting the spatial information. For comparison's sake we consider four cases, described in Table 4. The first case refers to DTW-FCMd clustering method described in D'Urso et al. (2018). The second and the third cases are particular instances of the proposed DTW-FCMd-STT, in which we exploited only a part of the spatial information provided by the proximity matrices P_1 and P_2 (see Figure 6). In the fourth case, the spatial information is fully exploited.

To evaluate the classification capability, we used the Fuzzy Rand Index (FRI) proposed by Hüllermeier et al. (2012), comparing the fuzzy partition obtained with the theoretical crisp reference partition. The closer FRI is to 1, the closer the fuzzy partition to the theoretical crisp reference partition. The results of the simulation are reported in the last column of Table 4. DTW-FCMd provides a partition that takes into account only the time dimension, which is rather fuzzy as explained, thus explaining the relative low value of FRI (case A). Exploiting only a part of the spatial information slightly enhances the classification capability of DTW-FCMd-STT with respect to DTW-FCMd (cases B and

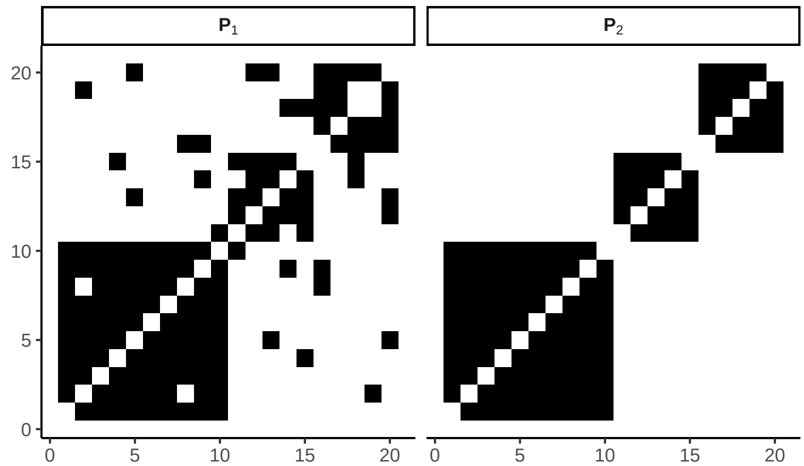


Figure 6: Proximity matrices – black squares indicate the proximity between two generic units (simulation study 3)

Case	Method	P_1	P_2	β_1	β_2	FRI
A	DTW-FCMd	No	No	0.0	0.0	0.720
B	DTW-FCMd-STT	Yes	No	1.8	0.0	0.734
C	DTW-FCMd-STT	No	Yes	0.0	1.8	0.741
D	DTW-FCMd-STT	Yes	Yes	1.8	1.8	0.985

Table 4: Fuzzy Rand Indices for simulation study 3, according to different clustering models (row wise) and different settings of spatial parameters (column wise)

C). On the contrary, by exploiting the whole spatial information, the clustering method is capable to correctly identify the clustering structure of the data at hand, properly incorporating the spatial information (case D). This evidence is further corroborated by the membership degrees obtained in the four cases, illustrated by the ternary plots[1] reported in Figure 7. In the ternary plot, every point represents the membership degrees of the corresponding time series in the three cluster. The more a point is close to a vertex of the triangle, the less uncertain is the assignment of the time series to the corresponding cluster.

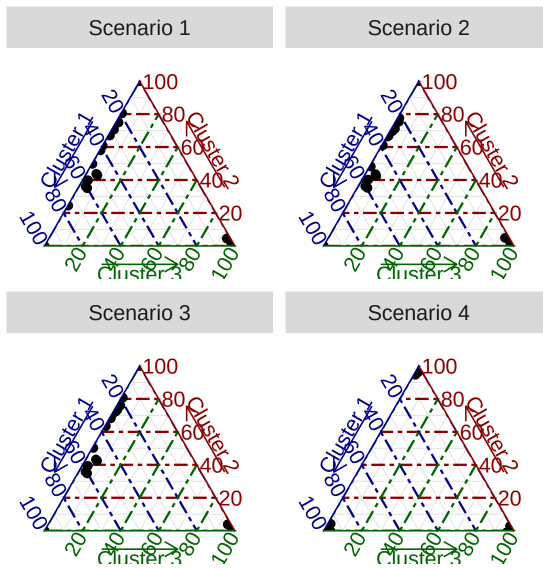


Figure 7: Membership degrees (simulation study 3)

As a final word, it should be stressed that the purpose of the present simulation is to clarify how the spatial information is embedded into the proposed clustering method.

3.4. Simulation study 4

For this simulation study, we partly replicated a simulation study proposed by D'Urso (2005a) and D'Urso et al. (2018) with an artificial dataset characterised by three well-separated clusters of four, three, and three multivariate time trajectories, respectively, and one outlier time trajectory (Figure 8). The length of each time series simulated is $T = 6$. The proximity matrix in Figure 9 represents the spatial component that has been included in this simulation study. Notice that a black square indicates proximity between units i and i', while a red square indicates proximity between an outlier and a generic unit.

Being the time series of the same length and having added only one proximity matrix, DTW-FCMD (D'Urso et al., 2018), our proposed clustering method (DTW-FCMd-STT), CS-FCM (D'Urso, 2005a), and CS-FCM-STT (Coppi et al., 2010) are fully comparable.

[1]The ternary plots have been produced by means of the R package `ggtern` (Hamilton & Ferry, 2018).

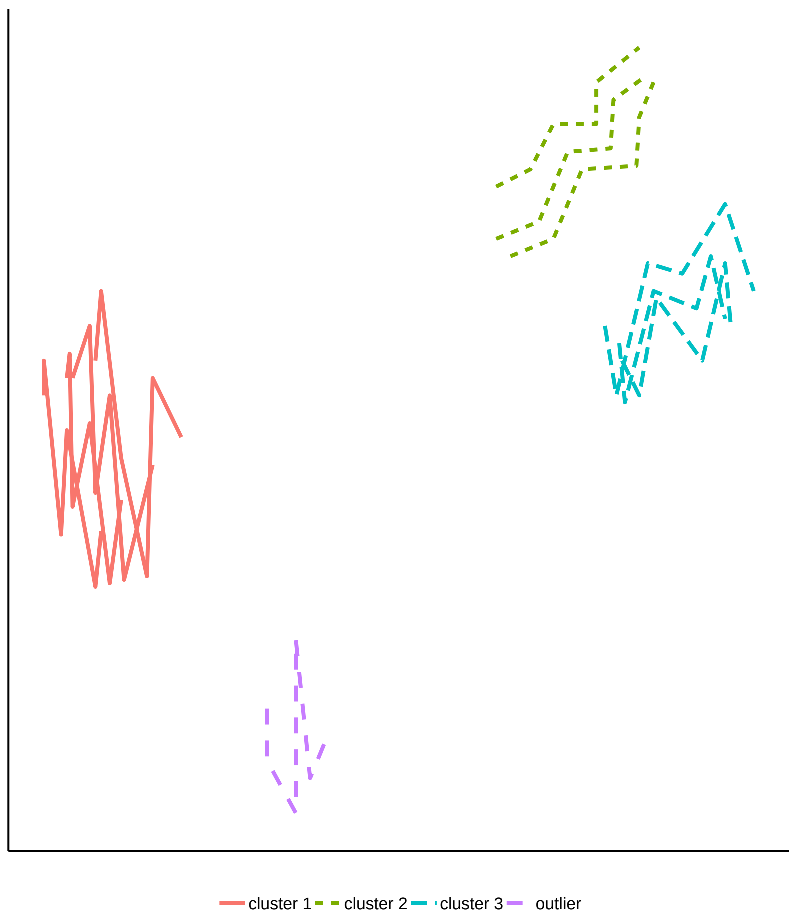


Figure 8: Simulated data for simulation study 4. Data are generated to be classified into three well separated clusters and one outlier time series. Time series belonging to different clusters and the outliers are depicted with different colours and line types

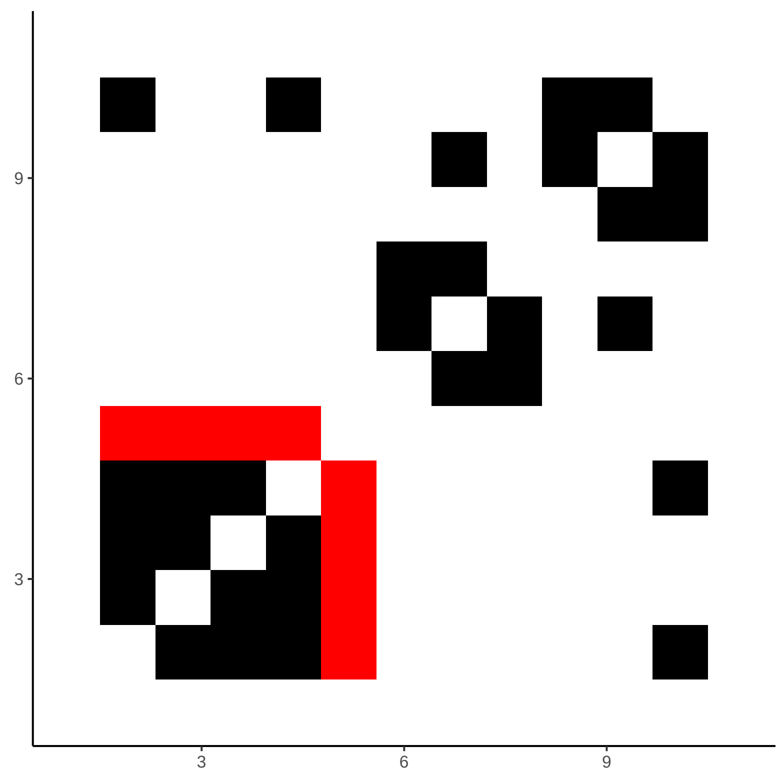


Figure 9: Proximity matrix – black squares indicate the proximity between two generic units: red squares indicate the proximity between the outlier and the corresponding unit (simulation study 4)

Case	Method	Outlier	Spatial information	FRI
A	DTW-FCMd	No	No	0.984
B		Yes	No	0.797
C	DTW-FCMd-STT	No	Yes	0.978
D		Yes	Yes	0.978
E	CS-FCM	No	No	0.990
F		Yes	No	0.780
G	CS-FCM-STT	No	Yes	0.948
H		Yes	Yes	0.761

Table 5: Fuzzy Rand Indices for simulation study 4, according to different clustering models (row wise) and to the presence of spatial information and/or the outlier time series (column wise)

Therefore, the simulation study is aimed to compare the classification capability of the above mentioned methods. Implicitly, we also compare DTW-FCMd-STT and CS-FCM-STT in the way they exploit the spatial information, in particular in the presence of a slight disturbance, given by the outlier time series. The value of β for both DTW-FCMd-STT and CS-FCM-STT has been set to 1.

In Table 5, FRI values for the different cases examined are reported. As expected, when the outlier time series is dropped from data, all clustering methods display a very good clustering performance. On the contrary, only DTW-FCMD-STT is able to resist to the presence of one outlier in the dataset.

4. Illustration with economic data

4.1. Study data

In this analysis, we consider annual tourist arrivals in the municipalities located in South-Tyrol region (Northern Italy) collected by ASTAT (the local institute of statistics) from 2008 to 2014. Given a geographic region having various localities as possible tourist attractions, we aim at identifying agglomerations of cities characterised by a common trend of the tourist flows over time taking into account the particular geographical and political underlined structure. South–Tyrol is in fact a tourist destination characterised by 116 municipalities grouped into eight administrative districts that follow the geomorphology of the region (see Figure 10).

Therefore, each municipality is characterised by two spatial information: whether two units are contiguous or not; whether two units belong to the same district or not. In this paper, each municipality is described by the annual time series on tourist flows from the two main markets, i.e. Germany and Italy (domestic tourists). Table 6 shows the descriptive characteristics of the two time series highlighting the high variability of arrivals among municipalities in each year observed.

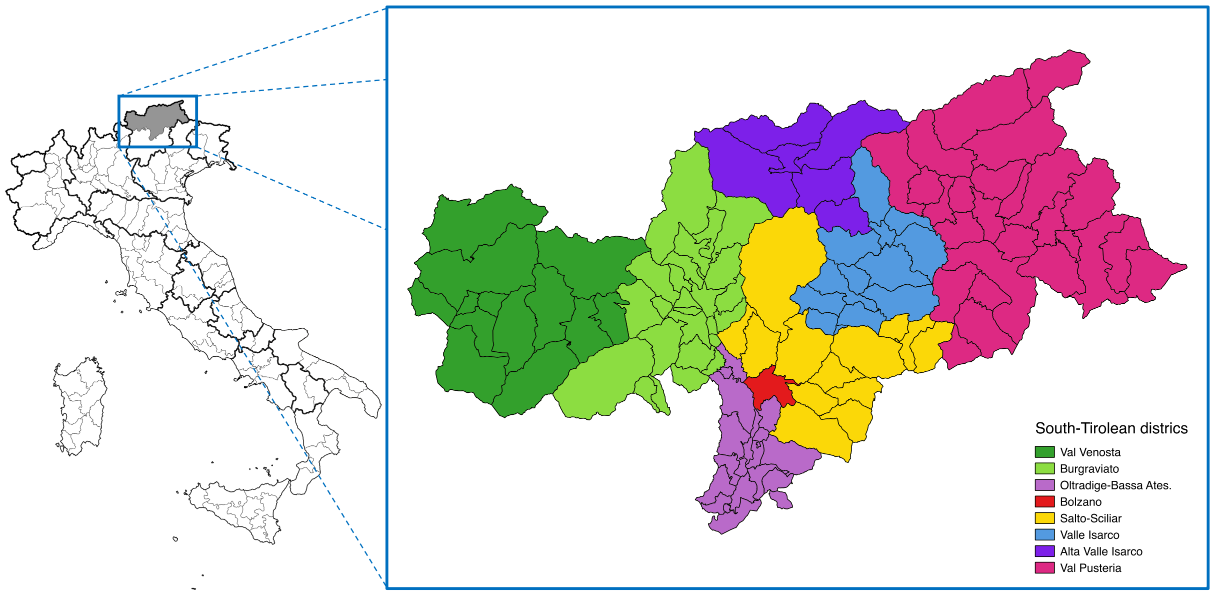


Figure 10: South–Tyrol region

	Mean		SD		MIN		MAX	
Year	Germany	Italy	Germany	Italy	Germany	Italy	Germany	Italy
2008	19843.34	18198.87	20058.15	25493.66	0	0	103026	109185
2009	20334.71	18849.07	20735.57	26094.45	0	0	106228	113199
2010	21005.07	18997.12	21771.90	26757.23	0	2	111202	115211
2011	21901.66	18952.86	22375.17	26338.25	0	2	114095	112591
2012	23066.78	18774.24	23464.45	25751.07	0	2	117825	113070
2013	23300.02	18188.68	23779.26	25245.33	0	3	117064	110082
2014	23889.45	18024.78	23973.20	24675.21	0	0	111843	111070

Table 6: Descriptive statistics of annual tourist arrivals from Germany and Italy

As highlighted in Figure 11, units are spatially autocorrelated, especially with regards to domestic tourists who are mainly grouped in Val Pusteria (East part of the region).

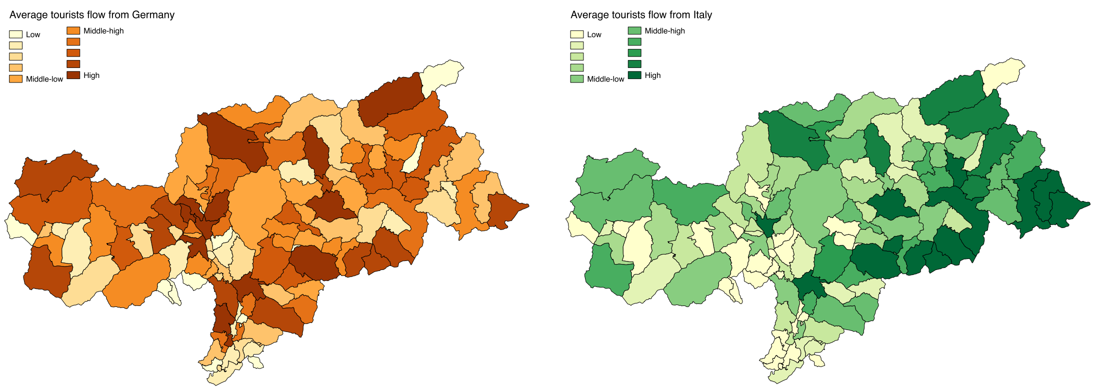


Figure 11: Average annual tourist flows

By means of the suggested DTW-FCMd-STT clustering algorithm with spatial penalty terms, we have the opportunity to: 1) identify agglomerations of cities characterised by similar tourist arrival trends, by considering units' geographical proximity and district memberships; 2) recognise the medoid of each agglomeration, i.e. the municipality that characterises each agglomeration and that can be considered as the representative touristic municipality (in statistical terms) of a given sub-region.

4.2. Clustering results

The optimal iterative solution is obtained by solving the DTW-FCMd-STT algorithm with the Lagrangian multipliers method where:

(1) the fuzziness parameter has been fixed to $m = 1.5$ (Kamdar & Joshi, 2000);
(2) the optimal number of clusters C of the DTW-FCMd-STT algorithm without penalty terms has been identified by means of the fuzzy cluster validity measures presented in section 2.4;
(3) the values of the two spatial penalty coefficients (i.e. β_1 and β_2) have been selected in order to maximize the multivariate spatial autocorrelation of the whole area (without considering the possible clustering structure) when both proximity matrices are considered.

Figure 12 summarises the values of the FS and XB indices calculated for any partition C from 2 to 9 when the spatial penalty terms are not included in the WTD-FCMd clustering algorithm. The trajectories of the two indices suggest that the best partitions are $C = 2$ followed by the four and six clusters partitions.

The weighted multivariate spatial autocorrelation of the whole area has been computed by means of equation 10 imposing $\bar{\mathbf{X}}$ equals to the identity matrix. The weighting spatial matrix $\tilde{\mathbf{P}}$ is computed through equation 11 fixing $K = 2$:

$$\tilde{\mathbf{P}} = w_1\mathbf{P}_1 + w_2\mathbf{P}_2$$

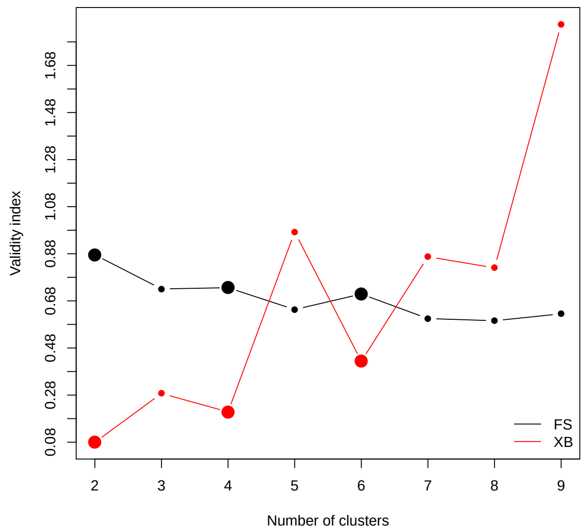


Figure 12: FS and XB validity index values for each cluster partition C from 2 to 9

where $\mathbf{P}_1$ is a non-negative (116×116) data matrix, whose generic entry $p_{1ii'}$ can be interpreted as the spatial proximity between the i-th and i'-th units $(i, i' = 1, \ldots, 116)$, $\mathbf{P}_2$ is another non-negative (116×116) data matrix, whose generic entry $p_{2ii'}$ describes whether the i-th and i'-th units belong to the same district or not, $w_1 = 1 - w_2$ is the parameter to be identified in order to maximize the weighted multivariate spatial correlation. Once the optimal value of w_1, i.e. w_1^*, is identified, we suggest to define the two spatial penalty parameters, i.e. β_1 and β_2, such as $w_1 = \frac{\beta_1}{\beta_1+\beta_2}$. Consequently, the best combination of β_1 and β_2 will be the one that allows to obtain the closer value to w_1^*. In this way we guarantee that the higher w_1, the higher β_1, i.e. the two parameters related to the same proximity matrix $\mathbf{P}_1$ go on the same direction. In this study, the maximum value of the weighted multivariate spatial autocorrelation for the whole area is 0.21, indicating a positive spatial autocorrelation between observed municipalities in inbound tourist from Germany and domestic tourist flows, and $w_1^* = 0.68$, as represented in Figure 13.

In the following, we will concentrate our attention on the four-clusters and six-clusters solutions. In fact, from a managerial and practical perspective, the two-clusters is not an appealing solution since it is not generally informative and useful to draw new policies and strategies.

Fixing $C = 4$, the best combination of β_1 and β_2, i.e. the one that allows to maximize the weighted multivariate spatial autocorrelation, is $\beta_1 = 0.01$ and $\beta_2 = 0.005$, which allows to obtain a fairly high spatial autocorrelation between geographical units ($FM = 0.50$). Comparing the final 4 clusters obtained with and without the two spatial proximity matrices, it emerges that the spatial information allows making small adjustments to the membership degrees of the final matrix but not severe changes in the final fuzzy cluster partition.

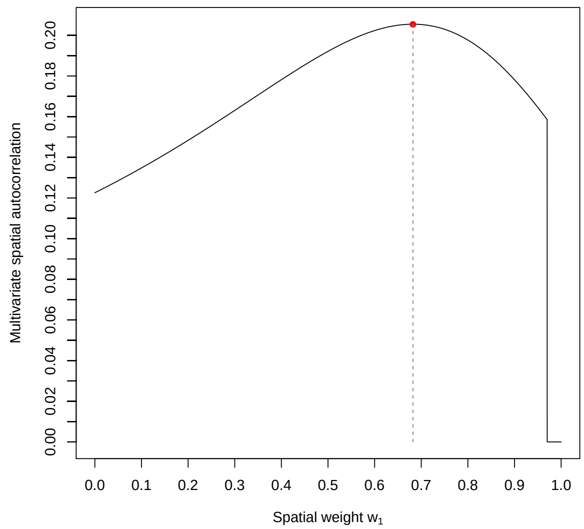


Figure 13: Values of the multivariate spatial autocorrelation of the whole area when proximity matrices are considered

Conversely, when $C = 6$ the best combination of β_1 and β_2 is $\beta_1 = 0.61$ and $\beta_2 = 0.32$, which allows to obtain a fairly high spatial autocorrelation between geographical units ($FM = 0.47$). As in the previous configuration, the proximity between areas is more relevant than the belonging to the same district. Figure 14 compares the membership degrees of each unit computed using DTW-FCMd-STT with and without penalty terms. The most evident changes, both in terms of intensity and frequency, are observable in cluster 1, 4, and 5. A similar conclusion can be reached observing the fuzzy cluster size, i.e. the sum of membership degrees per cluster, represented in Table 7. This measure is a proxy of the cluster size usually gather from crisp algorithm and it allows to spot both niches (as cluster 2 and 6) and bigger clusters (as cluster 4 and 5). Overall, cluster 1, 4, and 5 are the biggest clusters that highlight also the biggest changes.

	1	2	3	4	5	6
Without spatial terms	10.7047	6.22686	9.02877	60.9413	22.6855	6.41291
With spatial terms	11.5986	6.19582	9.04499	61.2298	21.5608	6.37000

Table 7: Sum of the membership degrees by cluster

For a deeper understanding and interpretation of the differences between the results of the two clustering algorithms, the membership degrees of each town/village, along with the medoids of each cluster, are represented in Figure 15.

The final membership degrees to cluster 1, 4, and 5 obtained excluding and including the penalty terms are compared to point out the most relevant changes. It is worthy of

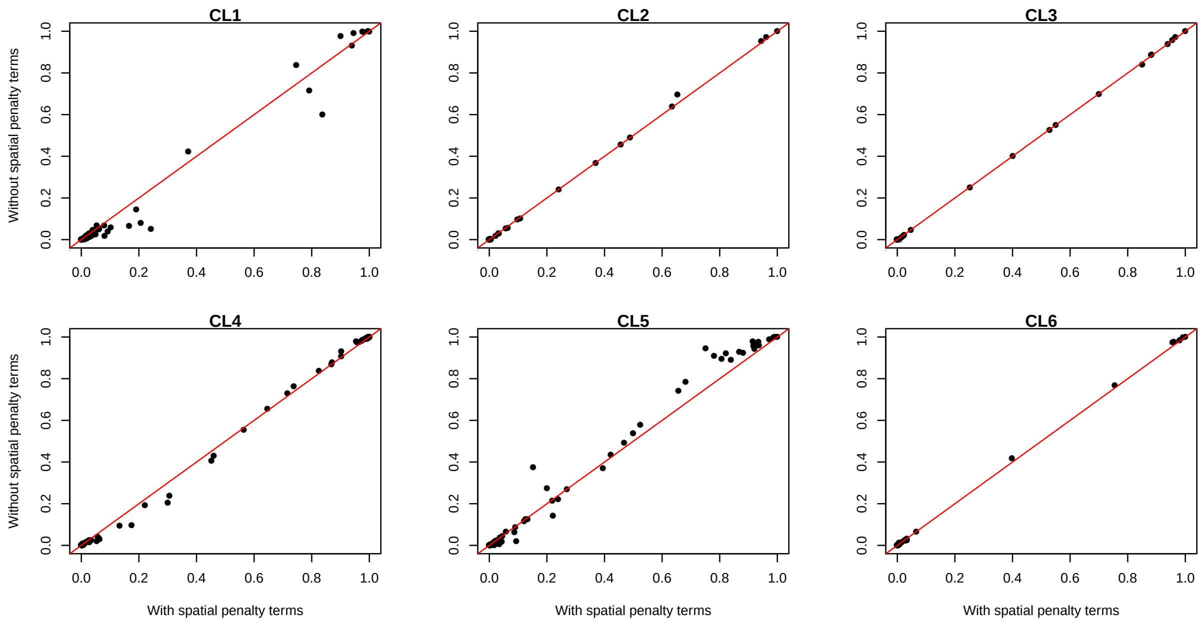


Figure 14: Comparing unit membership degrees to each cluster obtained using DTW-FCMd-STT with and without spatial terms

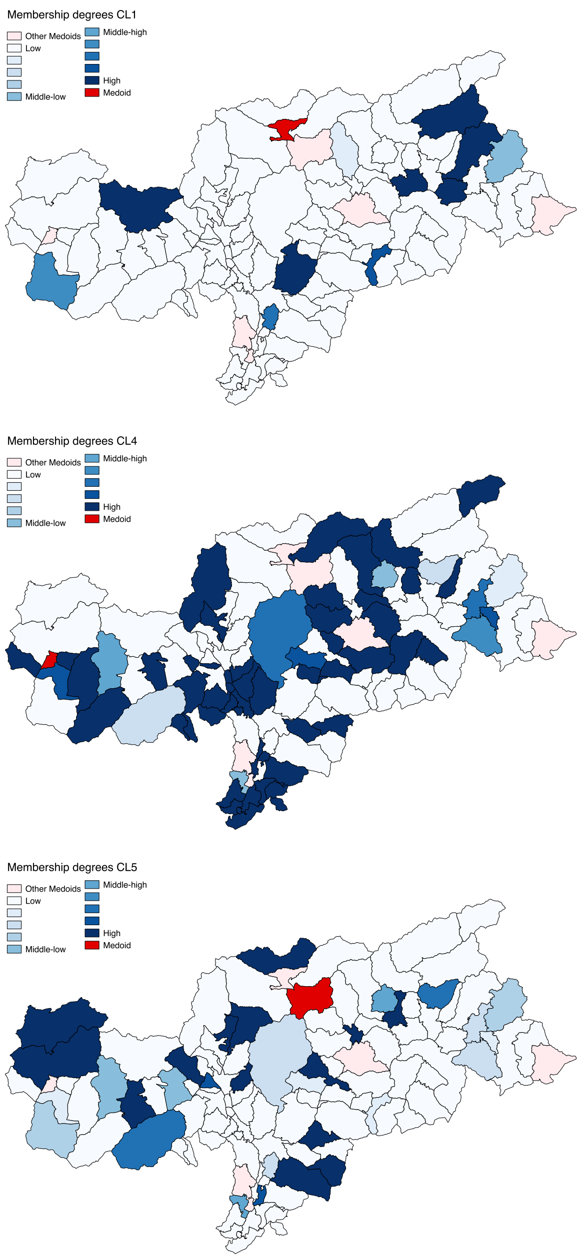
notice that the inclusion of the penalty terms in the clustering algorithm does not force final clusters to be made by neighbours town/village or to recall the districts. The change in the medoid of cluster 1 is the most noticeable and important change observable. This result have important practical consequences when policies and strategies are made at an aggregate (medoid) level instead of at a municipality (geographical unit) level.

For instance, marketing and promotional strategies to attract and host domestic or German tourists will be different depending on the decision to include or not the penalty terms (see Figure 16a). Furthermore, in Figure 16b the average cluster time series of the tourist flows coming from Germany and Italy are represented. Tourist flows are unchanged (domestic tourist) or slightly change (tourist from Germany) for cluster 2, 3, and 6, while the remaining clusters present more consistent variations, especially for tourists coming from Germany.

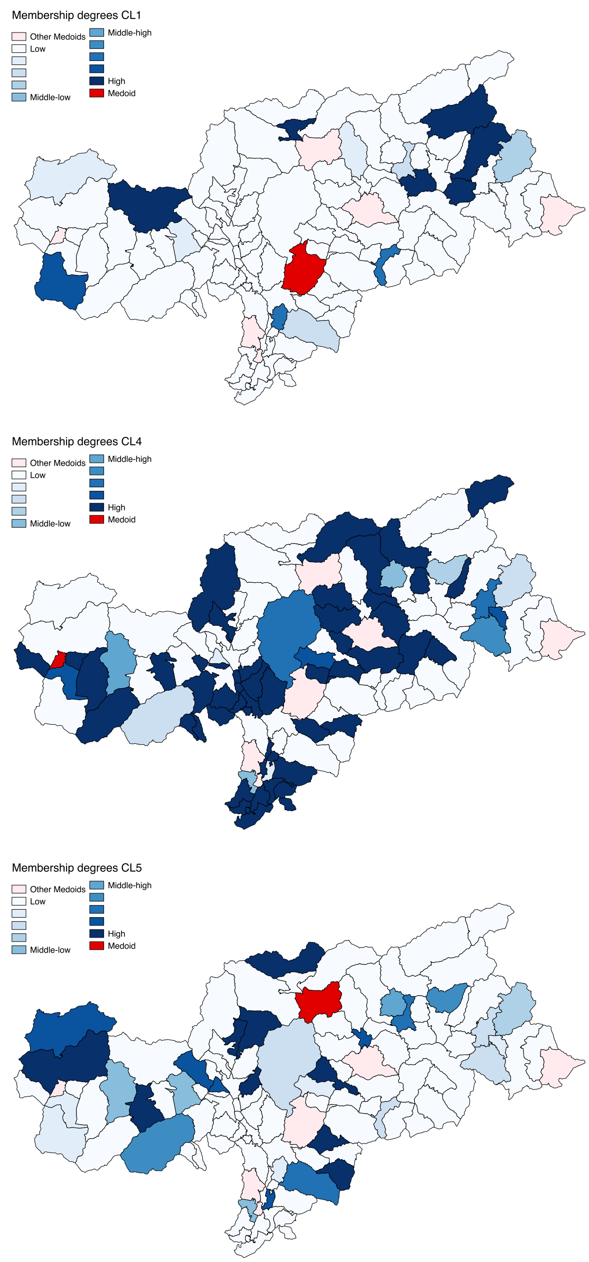
Therefore, due to the particular geographical and political structure of the region, ignoring the two proximity levels may lead to incorrect results and policies.

5. Conclusions

In this paper, the Dynamic Time Warping Fuzzy C-Medoids for Spatial-Temporal Trajectories (DTW-FCMd-STT) clustering algorithm with penalty terms, a new clustering algorithm for the classification of units described by both multivariate time series and spatial information, has been introduced. In particular, the main aim of this study is to present a multivariate generalisation of the Coppi et al. (2010) clustering algorithm by 1) adopting a more flexible distance measure, the DTW dissimilarity measure, and 2) extending the



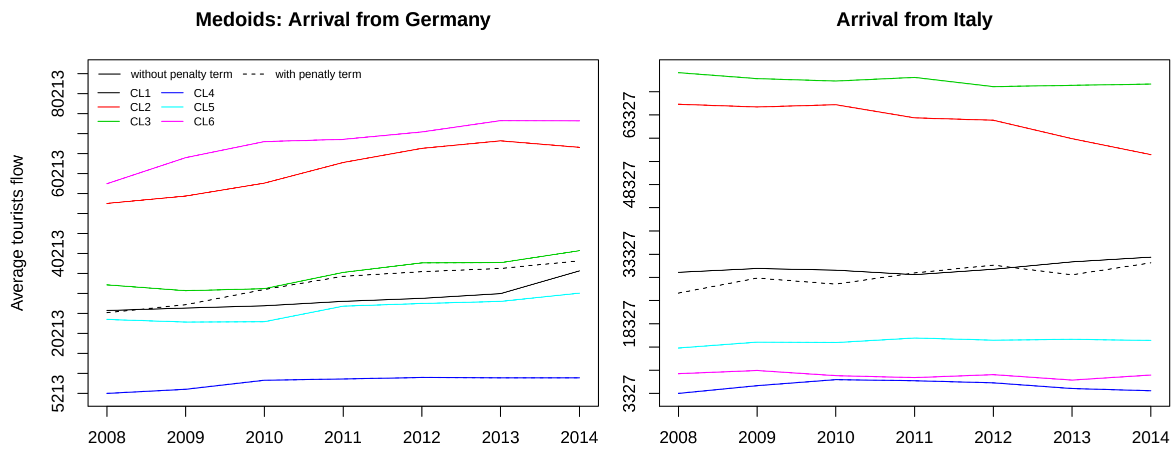
(a) Without spatial terms



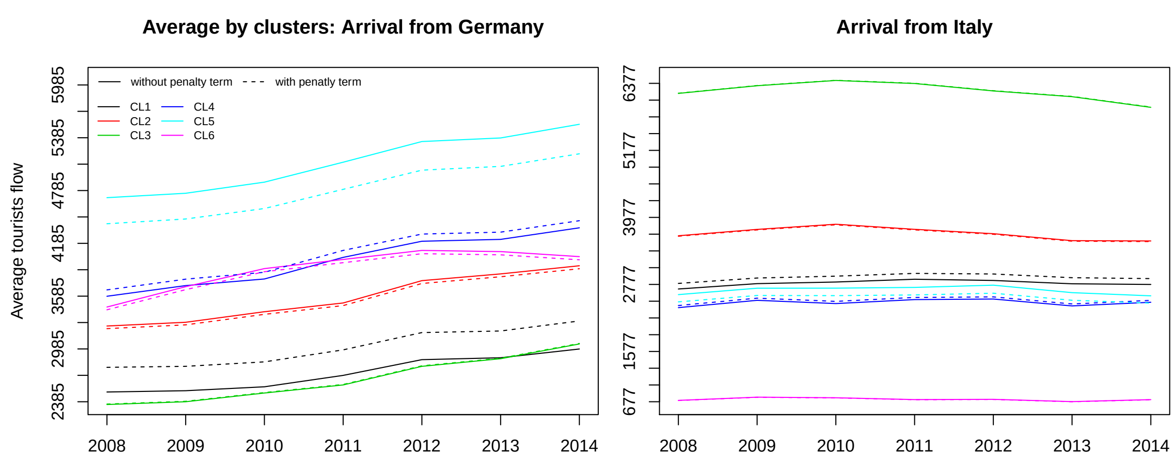
(b) With spatial terms

Figure 15: DTW-FCMd-STT without (on the left) and with (on the right) spatial terms when $C = 6$

Figure 16: Medoids time series (16a) and weighted average arrivals by cluster (16b)



(a)



(b)

possibility to classify units on which either different kinds or different levels of proximity are identifiable. Furthermore, a new weighted multivariate spatial autocorrelation index to evaluate the autocorrelation of the final fuzzy partition, i.e. the Fuzzy Moran's index, has been defined and presented.

Different simulation studies and a real dataset drawn by the tourism field have been presented to illustrate the usefulness and effectiveness of the suggested clustering method for spatial-temporal series. In particular, the findings of the simulation studies describe the sensitivity of the DTW-FCMd-STT clustering algorithm to changes in the proximity matrices. The application to the real case study shows that the DTW-FCMd-STT algorithm may help in the identification of groups that are spatially close, making more appealing the applicability of the results of the cluster analysis. Furthermore, the Fuzzy Moran's index reveal that a fairly high spatial autocorrelation between geographical units exists. Consequently, this result also indicate the presence of a positive spill-over effect among municipalities, i.e. one municipality's tourism industries affects the tourism flows of neighbours municipalities due to the existence of spatial externalities.

Finally, it is worth exploring also the possibility of obtaining more robust version of the proposed clustering algorithm, in order to cope with the presence of noise both in the time and in the spatial dimensions.

Acknowledgment

The authors thank the Editor and the referees for their useful comments and suggestions which helped to improve the quality and presentation of this manuscript.

References

Ahmed, M. N., Yamany, S. M., Mohamed, N., Farag, A. A., & Moriarty, T. (2002). A modified fuzzy c-means algorithm for bias field estimation and segmentation of mri data. *IEEE Transactions on Medical Imaging*, *21*, 193–199.

Alonso, A., & Maharaj, E. (2006). Comparison of time series using subsampling. *Comput. Stat. Data Anal.*, *50*, 2589–2599.

Basford, K., & McLachlan, G. (1985). The mixture method of clustering applied to three-way data. *J. Classification*, *2*, 109–125.

Bazin, P.-L., & Pham, D. L. (2007). Topology-preserving tissue classification of magnetic resonance brain images. *IEEE Transactions on Medical Imaging*, *26*, 487–496.

Begum, N., Ulanova, L., Wang, J., & Keogh, E. (2015). Accelerating dynamic time warping clustering with a novel admissible pruning strategy. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 49–58). ACM.

Berndt, D. (1994). Using Dynamic Time Warping to find patterns in time series. In *AAAI-94 Workshop on Knowledge Discovery in Databases*.

Bezdek, J. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Norwell, MA, USA: Kluwer Academic.

Birant, D., & Kut, A. (2007). ST-DBSCAN: An algorithm for clustering spatial-temporal data. *Data Knowl Eng.*, *60*, 208–221.

Caiado, J., & Crato, N. (2010). Identifying common dynamic features in stock returns. *Quantitative Finance*, *10*, 797–807.

Caiado, J., Crato, N., & Peña, D. (2009). Comparison of times series with unequal length in the frequency domain. *Communications in Statistics - Simulation and Computation*, *38*, 527–540.

Caiado, J., Crato, N., & Peña, D. (2006). A periodogram-based metric for time series classification. *Computational Statistics & Data Analysis*, *50*, 2668–2684.

Caiado, J., Maharaj, E. A., & D'Urso, P. (2015). Time series clustering. In C. Hennig, M. Meila, F. Murtagh, & R. Rocci (Eds.), *Handbook of Cluster Analysis* (pp. 241–264). Chapman and Hall/CRC.

Campello, R. J., & Hruschka, E. R. (2006). A fuzzy extension of the silhouette width criterion for cluster analysis. *Fuzzy Sets and Systems*, *157*, 2858–2875.

Castagno, R., Ebrahimi, T., & Kunt, M. (1998). Video segmentation based on multiple features for interactive multimedia applications. *IEEE Transactions on Circuits and Systems for Video Technology*, *8*, 562–571.

Chatzis, S. P., & Varvarigou, T. A. (2008). A fuzzy clustering approach toward hidden Markov random field models for enhanced spatially constrained image segmentation. *IEEE Transactions on Fuzzy Systems*, *16*, 1351–1361.

Chuang, K.-S., Tzeng, H.-L., Chen, S., Wu, J., & Chen, T.-J. (2006). Fuzzy c-means clustering with spatial information for image segmentation. *Computerized medical imaging and graphics*, *30*, 9–15.

Coppi, R., & D'Urso, P. (2002). Fuzzy k-means clustering models for triangular fuzzy time trajectories. *Statistical Methods and Applications*, *11*, 21–40.

Coppi, R., & D'Urso, P. (2003). Three-way fuzzy clustering models for LR fuzzy time trajectories. *Computational Statistics & Data Analysis*, *43*, 149–177.

Coppi, R., & D'Urso, P. (2006). Fuzzy unsupervised classification of multivariate time trajectories with the Shannon entropy regularization. *Computational Statistics & Data Analysis*, *50*, 1452–1477.

Coppi, R., D'Urso, P., & Giordani, P. (2010). A fuzzy clustering model for multivariate spatial time series. *Journal of Classification*, *27*, 54–88.

Di Nola, A., Loia, V., & Staiano, A. (2000). Genetic-based spatial clustering. In *IEEE International Conference on Fuzzy Systems* (pp. 953–956). volume 2.

Ding, H., Trajcevski, G., Scheuermann, P., Wang, X., & Keogh, E. (2008). Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, *1*, 1542–1552.

Disegna, M., D'Urso, P., & Durante, F. (2017). Copula-based fuzzy clustering of spatial time series. *Spatial Statistics*, *21*, 209–225.

D'Urso, P. (2000). Dissimilarity measures for time trajectories. *Statistical Methods & Applications*, *9*, 53–83.

D'Urso, P. (2004). Fuzzy C-means clustering models for multivariate time-varying data: Different approaches. *Internat. J. Uncertain. Fuzziness Knowledge-Based Systems*, *12*, 287–326.

D'Urso, P. (2005a). Fuzzy clustering for data time arrays with inlier and outlier time trajectories. *IEEE Transactions on Fuzzy Systems*, *13*, 583–604.

D'Urso, P. (2005b). Fuzzy clustering models for multivariate spatial-time series. In *3rd World Conference on Computational Statistics and Data Analysis*.

D'Urso, P. (2015). Fuzzy clustering. In C. Hennig, M. Meila, F. Murtagh, & R. Rocci (Eds.), *Handbook of Cluster Analysis* (pp. 545–574). Chapman & Hall.

D'Urso, P., Cappelli, C., Di Lallo, D., & Massari, R. (2013a). Clustering of financial time series. *Physica A: Statistical Mechanics and its Applications*, *392*, 2114–2129.

D'Urso, P., De Giovanni, L., Maharaj, E. A., & Massari, R. (2014). Wavelet-based Self-Organizing Maps for classifying multivariate time series. *Journal of Chemometrics*, *28*, 28–51.

D'Urso, P., De Giovanni, L., & Massari, R. (2015). Time series clustering by a robust autoregressive metric with application to air pollution. *Chemometrics and Intelligent Laboratory Systems*, *141(Supplement C)*, 107–124.

D'Urso, P., De Giovanni, L., & Massari, R. (2016). GARCH-based robust clustering of time series. *Fuzzy Sets and Systems*, *305*, 1–28.

D'Urso, P., De Giovanni, L., & Massari, R. (2018). Robust fuzzy clustering of multivariate time trajectories. *International Journal of Approximate Reasoning*, *99*, 12–38.

D'Urso, P., Di Lallo, D., & Maharaj, E. A. (2013b). Autoregressive model-based fuzzy clustering and its application for detecting information redundancy in air pollution monitoring networks. *Soft Comput.*, *17*, 83–131.

D'Urso, P., & Maharaj, E. (2009). Autocorrelation-based fuzzy clustering of time series. *Fuzzy Sets and Systems*, *160*, 3565–3589.

D'Urso, P., & Maharaj, E. A. (2012). Wavelets-based clustering of multivariate time series. *Fuzzy Sets and Systems*, *193*, 33–61.

D'Urso, P., Maharaj, E. A., & Alonso, A. M. (2017). Fuzzy clustering of time series using extremes. *Fuzzy Sets and Systems*, *318*, 56–79.

D'Urso, P., Manca, G., Waters, N., & Girone, S. (2019). Visualizing regional clusters of Sardinia's EU supported agriculture: A Spatial Fuzzy Partitioning Around Medoids. *Land Use Policy*, *83*, 571–580.

D'Urso, P., Massari, R., Cappelli, C., & De Giovanni, L. (2017). Autoregressive metric-based trimmed fuzzy clustering with an application to pm_{10} time series. *Chemometrics and Intelligent Laboratory Systems*, *13*, 583–604.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* KDD'96 (pp. 226–231). AAAI Press.

Estivill-Castro, V., & Yang, J. (2004). Fast and robust general purpose clustering algorithms. *Data Mining and Knowledge Discovery*, *8*, 127–150.

Everitt, B., Landau, S., & Leese, M. (2001). *Cluster Analysis*. (Forth ed.). London: Arnold Press.

Fouedjio, F. (2016). A hierarchical clustering method for multivariate geostatistical data. *Spatial Statistics*, *18*, 333–351.

Gao, X., & Yu, F. (2016). Fuzzy C-means with spatiotemporal constraints. *Proceedings - 2016 IEEE International Symposium on Computer, Consumer and Control, IS3C 2016*, (pp. 337–340).

García-Escudero, L., & Gordaliza, A. (1999). Robustness properties of k means and trimmed k means. *Journal of the American Statistical Association*, *94*, 956–969.

García-Escudero, L. A., Gordaliza, A., Matrán, C., & Mayo-Iscar, A. (2010). A review of robust clustering methods. *Advances in Data Analysis and Classification*, *4*, 89–109.

Geman, S., & Geman, D. (1987). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. In *Readings in Computer Vision* (pp. 564–584). Elsevier.

Giorgino, T. (2009). Computing and visualizing dynamic time warping alignments in R: the dtw package. *Journal of Statistical Software*, *31*, 1–24.

Gittleman, J., & Kot, M. (1990). Adaptation: Statistics and a null model for estimating phylogenetic effects. *Systematic Zoology*, *39*, 227–241.

Gordon, A. D. (1999). *Classification, (Chapman & Hall/CRC Monographs on Statistics & Applied Probability)*. Chapman and Hall/CRC.

Gordon, A. D., & Vichi, M. (2001). Fuzzy partition models for fitting a set of partitions. *Psychometrika*, *66*, 229–247.

Hamilton, N. E., & Ferry, M. (2018). ggtern: Ternary diagrams using ggplot2. *Journal of Statistical Software, Code Snippets*, *87*, 1–17. doi:`10.18637/jss.v087.c03`.

Handl, J., Knowles, J., & Kell, D. (2005). Computational cluster validation in post-genomic data analysis. *Bioinformatics*, *21*, 3201–3212.

Hao, J., Zhao, J., & Li, M. (2010). Spatial continuity incorporated multi-attribute fuzzy clustering algorithm for blood vessels segmentation. *Science China Information Sciences*, *53*, 752–759.

Hathaway, R. J., & Bezdek, J. C. (1994). NERF c-means: Non-Euclidean relational fuzzy clustering. *Pattern recognition*, *27*, 429–437.

Hu, T., & Sung, S. (2006). A hybrid EM approach to spatial clustering. *Computational Statistics & Data Analysis*, *50*, 1188–1205.

Hüllermeier, E., Rifqi, M., Henzgen, S., & Senge, R. (2012). Comparing Fuzzy Partitions: A Generalization of the Rand Index and Related Measures. *IEEE T. Fuzzy Systems*, *20*, 546–556.

Hwang, H., DeSarbo, W., & Takane, Y. (2007). Fuzzy clusterwise generalized structured component analysis. *Psychometrika*, *72*, 181–198.

Ienco, D., & Bordogna, G. (2016). Fuzzy extensions of the DBScan clustering algorithm. *Soft Comput.*, *in press*, 1–12.

Izakian, H., Pedrycz, W., & Jamal, I. (2013). Clustering spatiotemporal data: An augmented fuzzy C-means. *IEEE Transactions of Fuzzy Systems*, *21*, 855–868.

Izakian, H., Pedrycz, W., & Jamal, I. (2015). Fuzzy clustering of time series data using dynamic time warping distance. *Engineering Applications of Artificial Intelligence*, *39*, 235–244.

James, G., & Sugar, C. (2003). Clustering for sparsely sampled functional data. *J. Amer. Statist. Assoc.*, *98*, 397–408.

Jeong, Y.-S., Jeong, M., & Omitaomu, O. (2011). Weighted dynamic time warping for time series classification. *Pattern Recogn.*, *44*, 2231–2240.

Kalpakis, K., Gada, D., & Puttagunta, V. (2001). Distance measures for effective clustering of ARIMA time-series. *Proceedings - IEEE International Conference on Data Mining, ICDM*, (pp. 273–280).

Kamdar, T., & Joshi, A. (2000). On creating adaptive web servers using weblog mining. In *Technical Report TR-CS-00-05* (pp. 1–19). Baltimore County: Department of Computer Science and Electrical Engineering, University of Maryland.

Kaufman, L., & Rousseeuw, P. (2005). *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons.

Krishnapuram, R., & Freg, C. (1992). Fitting an unknown number of lines and planes to image data through compatible cluster merging. *Pattern Recogn.*, *25*, 385–400.

Krishnapuram, R., Joshi, A., Nasraoui, O., & Yi, L. (2001). Low-complexity fuzzy relational clustering algorithms for web mining. *Fuzzy Systems, IEEE Transactions on*, *9*, 595–607.

Lafuente-Rego, B., & Vilar, J. (2016). Clustering of time series using quantile autocovariances. *Adv. Data Anal. Classif.*, *10*, 391–415.

Li, X., Zhang, T., & Qu, Z. (2008). Image segmentation using fuzzy clustering with spatial constraints based on markov random field via bayesian theory. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, *91*, 723–729.

Liew, A., Leung, S., & Lau, W. (2000). Fuzzy image clustering incorporating spatial continuity. *IEE Proceedings-Vision, Image and Signal Processing*, *147*, 185–192.

Liew, A.-C., Leung, S. H., & Lau, W. H. (2003). Segmentation of color lip images by spatial fuzzy clustering. *IEEE transactions on Fuzzy Systems*, *11*, 542–549.

Ma, T., Hong, T., & Zhang, H. (2015). Tourism spatial spillover effects and urban economic growth. *Journal of Business Research*, *68*, 74–80.

Maharaj, E. (1996). A significance test for classifying ARMA models. *J. Stat. Comput. Simul.*, *54*, 305–331.

Maharaj, E., & D'Urso, P. (2010). A coherence-based approach for the pattern recognition of time series. *Physica A*, *389*, 3516–3537.

Maharaj, E., & D'Urso, P. (2011). Fuzzy clustering of time series in the frequency domain. *Inform. Sci.*, *181*, 1187–1211.

Maharaj, E., D'Urso, P., & Galagedera, D. (2010). Wavelet-based Fuzzy Clustering of Time Series. *J. Classification*, *27*, 231–275.

McBratney, A., & Moore, A. (1985). Application of fuzzy sets to climatic classification. *Agricultural and forest meteorology*, *35*, 165–185.

Mure, S., Grenier, T., Guttmann, C. R., & Benoit-Cattin, H. (2016). Unsupervised time-series clustering of distorted and asynchronous temporal patterns. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on* (pp. 1263–1267). IEEE.

Oates, T., Firoiu, L., & Cohen, P. R. (1999). Clustering time series with hidden markov models and dynamic time warping. In *Proceedings of the IJCAI-99 workshop on neural, symbolic and reinforcement learning methods for sequence learning* (pp. 17–21). Citeseer.

Otranto, E. (2010). Identifying financial time series with similar dynamic conditional correlation. *Computational Statistics & Data Analysis*, *54*, 1–15.

Páez, A., & Scott, D. (2005). Spatial statistics for urban analysis: A review of techniques with examples. *GeoJournal*, *61*, 61–53.

Petitjean, F., Ketterlin, A., & Gançarski, P. (2011). A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition*, *44*, 678–693.

Pham, D. (2001). Spatial models for fuzzy clustering. *Computer Vision and Image Understanding*, *84*, 285–297.

Pham, D. L., & Prince, J. L. (1999). Adaptive fuzzy segmentation of magnetic resonance images. *IEEE transactions on medical imaging*, *18*, 737–752.

Piccolo, D. (1990). A distance measure for classifying ARIMA models. *J. Time Ser. Anal.*, *11*, 153–164.

Rakthanmanon, T., Campana, B., Mueen, A., Batista, G., Westover, B., Zhu, Q., Zakaria, J., & Keogh, E. (2012). Searching and mining trillions of time series subsequences under dynamic time warping. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 262–270). ACM.

Sato, M., & Sato, Y. (1994). On a multicriteria fuzzy clustering method for 3-way data. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, *02*, 127–142.

Sato, M., Sato, Y., & Jain, L. (1997). *Fuzzy Clustering Models and Applications*. Physica-Verlag.

Shekhar, S., Jiang, Z., Ali, R., Eftelioglu, E., Tang, X., Gunturi, V., & Zhou, X. (2015). Spatiotemporal data mining: A computational perspective. *ISPRS International Journal of Geo-Information*, *4*, 2306–2338.

Simbahan, G. C., & Dobermann, A. (2006). An algorithm for spatially constrained classification of categorical and continuous soil properties. *Geoderma*, *136*, 504–523.

Subudhi, B. N., Bovolo, F., Ghosh, A., & Bruzzone, L. (2014). Spatio-contextual fuzzy clustering with Markov random field model for change detection in remotely sensed images. *Optics & Laser Technology*, *57*, 284–292.

Tolias, Y. A., & Panas, S. M. (1998a). Image segmentation by a fuzzy clustering algorithm using adaptive spatially constrained membership functions. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *28*, 359–369.

Tolias, Y. A., & Panas, S. M. (1998b). On applying spatial constraints in fuzzy image clustering using a fuzzy rule-based system. *IEEE Signal Processing Letters*, *5*, 245–247.

Torabi, M. (2014). Spatial generalized linear mixed models with multivariate CAR models for areal data. *Spat. Statist.*, *10*, 12–26.

Torabi, M. (2016). Hierarchical multivariate mixture generalized linear models for the analysis of spatial data: An application to disease mapping. *Biom. J.*, *58*, 1138–1150.

Velichko, V., Sato, M., & Zagoruyko, N. (1970). Automatic recognition of 200 words. *International Journal of Man-Machine Studies*, *2*, 223–234.

Vilar, J. A., Lafuente-Rego, B., & D'Urso, P. (2017). Quantile autocovariances: a powerful tool for hard and soft partitional clustering of time series. *Fuzzy Sets and Systems*, (p. in press).

Viroli, C. (2011). Finite mixtures of matrix normal distributions for classifying three-way data. *Stat. Comput.*, *21*, 511–522.

Wang, M., Wang, A., & Li, A. (2006). Mining spatial-temporal clusters from geo-databases. In X. Li, O. R. Zaïane, & Z. Li (Eds.), *Advanced Data Mining and Applications: Second International Conference, ADMA 2006, Xi'an, China, August 14-16, 2006 Proceedings* Advanced Data Mining and Applications (pp. 263–270). Berlin, Germany: Springer.

Wang, Z. M., Soh, Y. C., Song, Q., & Sim, K. (2009). Adaptive spatial information-theoretic clustering for image segmentation. *Pattern Recognition*, *42*, 2029–2044.

Warren Liao, T. (2005). Clustering of time series data - a survey. *Pattern Recogn.*, *38*, 1857–1874.

Wedel, M., & Kamakura, W. A. (2000). *Market Segmentation: Conceptual and Methodological Foundations*. (2nd ed.). Dordrecht: Kluwer Academic Publishers.

Xie, J., Gao, H., Xie, W., Liu, X., & Grant, P. (2016). Robust clustering by detecting density peaks and assigning points based on fuzzy weighted K-nearest neighbors. *Inform. Sci.*, *354*, 19–40.

Xie, X., & Beni, G. (1991). A validity measure for fuzzy clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, *13*, 841–847.

Yang, M.-S., & Tsai, H.-S. (2008). A Gaussian kernel-based fuzzy c-means algorithm with a spatial bias correction. *Pattern recognition letters*, *29*, 1713–1725.

Yang, Y. (2012). Agglomeration density and tourism development in China: An empirical research based on dynamic panel data model. *Tourism Manage.*, *33*, 1347–1359.

Yang, Y., & Fik, T. (2014). Spatial effects in regional tourism growth. *Annals of Tourism Research*, *46*, 144–162.

Zhang, D. Q., & Chen, S. C. (2004). A novel kernelized fuzzy c-means algorithm with application in medical image segmentation. *Artificial Intelligence in Medicine*, *32*, 37–50.

Appendix A. Simulation studies

Appendix A.1. Simulation study 1

In this section, we report some further comments on the first simulation study.

The medoids and the fuzzy membership obtained are illustrated in Table A.8. The medoids' membership degrees are highlighted in bold. As we can observe, the medoids are units 3 and 7 over all the data configurations. Furthermore, in each data configuration the membership degrees of units 4 and 5 to each cluster decrease or increase alternating the greater weight between the contiguity matrices $\mathbf{P}_1$ and $\mathbf{P}_2$. In data configuration 4), where units 4 and 5 are closest, units 5 is in the same cluster of unit 4 when the weight of $\mathbf{P}_1$ is greater than the weight of $\mathbf{P}_2$ ($\beta_1 = 4$ and $\beta_2 = 0$; $\beta_1 = 8$ and $\beta_2 = 0$); in different clusters when the weight of $\mathbf{P}_2$ is greater than the weight of $\mathbf{P}_1$ ($\beta_1 = 0$ and $\beta_2 = 4$; $\beta_1 = 0$ and $\beta_2 = 8$). In data configuration 4) when $\beta_1 = 8$ and $\beta_2 = 0$ the clusters are (medoid in bold) (1, 2, **3**, 4, 5) and (6, **7**, 8); when $\beta_1 = 0$ and $\beta_2 = 8$ the clusters are (medoid in bold) (1, 2, **3**, 4) and (5, 6, **7**, 8).

The performance of the proposed clustering method measured by the Fuzzy Silhouette index FS—is described in Table A.9. As it can be seen, going from configuration 1) to 4) the value of the silhouette increases. In fact the medoids remain the same (3 and 7) and the $fuzzy$ units 4 and 5 decrease their membership to the natural clusters (1, 2, 3) and (7, 8, 9).

Appendix A.2. Simulation study 2

In this section, we report some further comments on the second simulation study.

The medoids and the fuzzy membership are illustrated in Tables A.10 and A.11. As we can observe, the medoids are units 3, 7, 9, 14 over the data configurations 1) and 2); units 4, 5, 12, 13 over almost all the data configurations 3) and 4). Table A.10 and A.11 show that in each data configuration the membership degrees of units 4, 5, 12, 13 to each cluster decrease or increase alternating the greater weight between $\mathbf{P}_1$ and $\mathbf{P}_2$. In data configuration 3), where units 4, 5, 12, 13 are getting closer: 1) when $\beta_1 = 20$ and $\beta_2 = 0$ the units 4, 5, 12, 13 are in the same cluster and the clusters are (medoid in bold): (1, 2, **3**), (6, **7**, 8), (4, 5, 9, 10, 11, **12**, 13), (**14**, 15, 16); 2) when $\beta_1 = 0$ and $\beta_2 = 20$ the units 4, 5, 12, 13 are in different clusters and the clusters are (medoid in bold) (1, 2, **3**, 4), (5, 6, **7**, 8), (**9**, 10, 11,12), (13, **14**, 15, 16).

In Table A.12 the main conclusions of the simulation study are reported. Notice that, going from configuration 1), 2) to 3), 4), the value of the silhouette decreases. In configurations 1), 2) the medoids are 3, 7, 9, 14; in configurations 3), 4) the medoids are almost always 4, 5, 12, 13 (the $fuzzy$ units). The performances get worse in data configuration 3) and 4) in relation to the increased similarity of the fuzzy units 4, 5, 12, 13. The best performance in data configuration 3) is (β_1, β_2)= (20, 0) where the medoids are 3, 7, 12, 14 and the high weight of $\mathbf{P}_1$ constraints the four fuzzy units in the same cluster (medoid 12).

$(\beta_1,\ \beta_2)$	(0, 0)		(4, 0)		(0, 4)		(8, 0)		(0, 8)	
	Data configuration 1)									
cluster	1	2	1	2	1	2	1	2	1	2
1	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
2	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
3	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**
4	0.7907	0.2093	0.8086	0.1914	0.8160	0.1840	0.8235	0.1765	0.8372	0.1628
5	0.3311	0.6689	0.3305	0.6695	0.2934	0.7066	0.3308	0.6692	0.2611	0.7389
6	0.0000	1.0000	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999
7	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**
8	0.0003	0.9997	0.0003	0.9997	0.0003	0.9997	0.0003	0.9997	0.0003	0.9997
	Data configuration 2)									
cluster	1	2	1	2	1	2	1	2	1	2
1	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
2	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
3	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**
4	0.7407	0.2593	0.7654	0.2346	0.7715	0.2285	0.7860	0.2140	0.7976	0.2024
5	0.3889	0.6111	0.3848	0.6152	0.3457	0.6543	0.3824	0.6176	0.3083	0.6917
6	0.0000	1.0000	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999
7	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**
8	0.0003	0.9997	0.0003	0.9997	0.0003	0.9997	0.0003	0.9997	0.0004	0.9996
	Data configuration 3)									
cluster	1	2	1	2	1	2	1	2	1	2
1	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
2	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
3	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**
4	0.6844	0.3156	0.7654	0.2346	0.7715	0.2285	0.7860	0.2140	0.7976	0.2024
5	0.4500	0.5500	0.3848	0.6152	0.3457	0.6543	0.3824	0.6176	0.3083	0.6917
6	0.0000	1.0000	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999
7	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**
8	0.0003	0.9997	0.0003	0.9997	0.0003	0.9997	0.0003	0.9997	0.0004	0.9996
	Data configuration 4)									
cluster	1	2	1	2	1	2	1	2	1	2
1	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
2	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001	0.9999	0.0001
3	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**
4	0.6271	0.3729	0.6853	0.3147	0.6689	0.3311	0.7168	0.2832	0.7051	0.2949
5	0.4874	0.5126	0.5256	0.4744	0.4603	0.5397	0.5169	0.4831	0.4138	0.5862
6	0.0000	1.0000	0.0000	1.0000	0.0001	0.9999	0.0000	1.0000	0.0001	0.9999
7	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**	**0.0000**	**1.0000**
8	0.0003	0.9997	0.0001	0.9999	0.0004	0.9996	0.0001	0.9999	0.0004	0.9996

Note: Medoids' membership degrees are in bold.

Table A.8: Membership degrees for simulation study 1, according to different combinations of β_1 and β_2 and data configurations

Data	(β_1, β_2)				
configuration	(0, 0)	(4, 0)	(0, 4)	(8, 0)	0, 8
1)	0.82	0.82	0.81	0.81	0.80
2)	0.82	0.82	0.81	0.81	0.80
3)	0.84	0.82	0.81	0.81	0.80
4)	0.90	0.89	0.83	0.89	0.80

Table A.9: Fuzzy Silhouette index values for simulation study 1 according to different setting of the parameters β_1, β_2 (column wise) and to data configuration (row wise)

(β_1, β_2)	(8, 0)				(0, 8)				(12, 0)				(0, 12)				(16, 0)				(0, 16)				(20, 0)				(0, 20)			
	Data configuration 1)																															
cluster	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	0.9989	0.0005	0.0001	0.0005	0.9990	0.0005	0.0001	0.0004	0.9989	0.0005	0.0001	0.0005	0.9991	0.0004	0.0001	0.0004	0.9989	0.0005	0.0001	0.0005	0.9992	0.0004	0.0001	0.0004	0.9989	0.0005	0.0001	0.0005	0.9992	0.0003	0.0001	0.0003
2	0.9990	0.0004	0.0001	0.0004	0.9991	0.0004	0.0001	0.0004	0.9990	0.0004	0.0001	0.0004	0.9992	0.0004	0.0001	0.0004	0.9990	0.0004	0.0001	0.0004	0.9992	0.0003	0.0001	0.0003	0.9990	0.0005	0.0001	0.0005	0.9993	0.0003	0.0001	0.0003
3	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**
4	0.8544	0.0608	0.0195	0.0654	0.9448	0.0232	0.0069	0.0250	0.8311	0.0698	0.0244	0.0746	0.9521	0.0200	0.0064	0.0215	0.8128	0.0767	0.0289	0.0816	0.9579	0.0175	0.0059	0.0186	0.7981	0.0821	0.0329	0.0869	0.9628	0.0154	0.0055	0.0163
5	0.0543	0.8801	0.0428	0.0229	0.0195	0.9577	0.0151	0.0077	0.0633	0.8571	0.0509	0.0287	0.0169	0.9628	0.0133	0.0070	0.0703	0.8384	0.0575	0.0338	0.0148	0.9670	0.0118	0.0064	0.0758	0.8228	0.0630	0.0384	0.0130	0.9705	0.0106	0.0059
6	0.0001	0.9997	0.0001	0.0000	0.0001	0.9997	0.0001	0.0000	0.0001	0.9997	0.0002	0.0000	0.0001	0.9997	0.0001	0.0000	0.0001	0.9996	0.0002	0.0001	0.0001	0.9997	0.0001	0.0000	0.0002	0.9996	0.0002	0.0001	0.0001	0.9998	0.0001	0.0000
7	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**
8	0.0009	0.9980	0.0008	0.0002	0.0008	0.9981	0.0008	0.0002	0.0009	0.9981	0.0008	0.0002	0.0008	0.9983	0.0007	0.0002	0.0008	0.9981	0.0008	0.0003	0.0007	0.9984	0.0007	0.0002	0.0008	0.9981	0.0008	0.0003	0.0007	0.9985	0.0006	0.0002
9	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**
10	0.0001	0.0004	0.9990	0.0004	0.0001	0.0004	0.9991	0.0004	0.0001	0.0004	0.9990	0.0004	0.0001	0.0003	0.9992	0.0004	0.0001	0.0004	0.9990	0.0005	0.0001	0.0003	0.9993	0.0003	0.0001	0.0005	0.9989	0.0005	0.0001	0.0003	0.9993	0.0003
11	0.0000	0.0001	0.9997	0.0002	0.0000	0.0001	0.9998	0.0001	0.0000	0.0001	0.9997	0.0002	0.0000	0.0001	0.9998	0.0001	0.0000	0.0001	0.9996	0.0002	0.0000	0.0001	0.9998	0.0001	0.0000	0.0001	0.9996	0.0002	0.0000	0.0001	0.9998	0.0001
12	0.0235	0.0689	0.8323	0.0753	0.0095	0.0298	0.9279	0.0328	0.0288	0.0773	0.8100	0.0839	0.0088	0.0258	0.9372	0.0282	0.0334	0.0836	0.7928	0.0902	0.0081	0.0226	0.9448	0.0245	0.0376	0.0885	0.7790	0.0949	0.0076	0.0199	0.9511	0.0215
13	0.0425	0.0149	0.0400	0.9026	0.0108	0.0035	0.0101	0.9755	0.0523	0.0200	0.0494	0.8783	0.0093	0.0032	0.0088	0.9787	0.0601	0.0247	0.0569	0.8583	0.0081	0.0030	0.0076	0.9813	0.0663	0.0290	0.0631	0.8416	0.0071	0.0027	0.0067	0.9835
14	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**
15	0.0002	0.0000	0.0002	0.9996	0.0002	0.0000	0.0002	0.9996	0.0002	0.0000	0.0002	0.9996	0.0002	0.0000	0.0002	0.9997	0.0002	0.0001	0.0002	0.9996	0.0001	0.0000	0.0001	0.9997	0.0002	0.0001	0.0002	0.9995	0.0001	0.0000	0.0001	0.9997
16	0.0001	0.0000	0.0002	0.9997	0.0001	0.0000	0.0001	0.9997	0.0001	0.0000	0.0002	0.9997	0.0001	0.0000	0.0001	0.9997	0.0001	0.0000	0.0002	0.9997	0.0001	0.0000	0.0001	0.9998	0.0001	0.0000	0.0002	0.9996	0.0001	0.0000	0.0001	0.9998
	Data configuration 2)																															
cluster	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	0.9987	0.0006	0.0001	0.0006	0.9989	0.0005	0.0001	0.0005	0.9986	0.0006	0.0001	0.0006	0.9989	0.0005	0.0001	0.0005	0.9986	0.0007	0.0001	0.0006	0.9990	0.0005	0.0001	0.0004	0.9985	0.0007	0.0002	0.0007	0.9991	0.0004	0.0001	0.0004
2	0.9989	0.0005	0.0001	0.0005	0.9990	0.0005	0.0001	0.0005	0.9988	0.0005	0.0001	0.0005	0.9990	0.0004	0.0001	0.0004	0.9987	0.0006	0.0001	0.0006	0.9991	0.0004	0.0001	0.0004	0.9987	0.0006	0.0002	0.0006	0.9992	0.0004	0.0001	0.0004
3	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**
4	0.7265	0.1104	0.0417	0.1214	0.8210	0.0729	0.0260	0.0801	0.7172	0.1130	0.0464	0.1234	0.8437	0.0632	0.0241	0.0690	0.7102	0.1148	0.0505	0.1245	0.8625	0.0553	0.0223	0.0599	0.7047	0.1160	0.0541	0.1252	0.8781	0.0487	0.0207	0.0525
5	0.1016	0.7662	0.0800	0.0522	0.0656	0.8520	0.0508	0.0316	0.1047	0.7537	0.0842	0.0574	0.0568	0.8695	0.0449	0.0288	0.1069	0.7438	0.0876	0.0618	0.0497	0.8842	0.0399	0.0263	0.1085	0.7357	0.0903	0.0655	0.0438	0.8965	0.0357	0.0240
6	0.0002	0.9996	0.0002	0.0001	0.0001	0.9997	0.0002	0.0000	0.0002	0.9996	0.0002	0.0001	0.0001	0.9997	0.0001	0.0000	0.0002	0.9995	0.0002	0.0001	0.0001	0.9997	0.0001	0.0000	0.0002	0.9995	0.0002	0.0001	0.0001	0.9997	0.0001	0.0000
7	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**
8	0.0010	0.9978	0.0009	0.0003	0.0009	0.9980	0.0008	0.0003	0.0010	0.9978	0.0009	0.0003	0.0008	0.9981	0.0008	0.0002	0.0010	0.9978	0.0009	0.0003	0.0008	0.9983	0.0007	0.0002	0.0010	0.9978	0.0009	0.0003	0.0007	0.9984	0.0007	0.0002
9	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**
10	0.0001	0.0005	0.9988	0.0005	0.0001	0.0004	0.9990	0.0005	0.0002	0.0006	0.9987	0.0006	0.0001	0.0004	0.9990	0.0004	0.0002	0.0006	0.9986	0.0006	0.0001	0.0004	0.9991	0.0004	0.0002	0.0006	0.9985	0.0006	0.0001	0.0004	0.9991	0.0004
11	0.0000	0.0001	0.9996	0.0002	0.0000	0.0001	0.9997	0.0002	0.0001	0.0002	0.9995	0.0003	0.0000	0.0001	0.9997	0.0002	0.0001	0.0002	0.9994	0.0003	0.0000	0.0001	0.9997	0.0002	0.0001	0.0002	0.9993	0.0004	0.0000	0.0001	0.9997	0.0002
12	0.0492	0.1237	0.6918	0.1353	0.0324	0.0866	0.7863	0.0947	0.0539	0.1250	0.6853	0.1358	0.0301	0.0755	0.8125	0.0819	0.0579	0.1257	0.6807	0.1358	0.0280	0.0663	0.8343	0.0715	0.0615	0.1260	0.6771	0.1354	0.0260	0.0586	0.8526	0.0628
13	0.0796	0.0342	0.0737	0.8125	0.0439	0.0177	0.0407	0.8977	0.0849	0.0394	0.0791	0.7967	0.0377	0.0161	0.0351	0.9111	0.0889	0.0440	0.0833	0.7839	0.0326	0.0147	0.0306	0.9221	0.0920	0.0481	0.0866	0.7733	0.0285	0.0135	0.0268	0.9312
14	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**
15	0.0002	0.0001	0.0002	0.9995	0.0002	0.0000	0.0002	0.9996	0.0002	0.0001	0.0002	0.9995	0.0002	0.0000	0.0002	0.9996	0.0002	0.0001	0.0002	0.9994	0.0002	0.0000	0.0002	0.9996	0.0003	0.0001	0.0003	0.9994	0.0001	0.0000	0.0001	0.9997
16	0.0001	0.0000	0.0002	0.9996	0.0001	0.0000	0.0002	0.9997	0.0002	0.0000	0.0002	0.9996	0.0001	0.0000	0.0001	0.9997	0.0002	0.0001	0.0002	0.9995	0.0001	0.0000	0.0001	0.9997	0.0002	0.0001	0.0002	0.9995	0.0001	0.0000	0.0001	0.9997

Note: Medoids' membership degrees are in bold.

Table A.10: Membership degrees for simulation study 2, according to different combinations of β_1 and β_2, and to data configurations 1) and 2)

(β_1, β_2)	(8, 0)				(0, 8)				(12, 0)				(0, 12)				(16, 0)				(0, 16)				(20, 0)				(0, 20)			
	Data configuration 3)																															
cluster	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	0.5570	0.1876	0.0923	0.1631	0.5570	0.1876	0.0923	0.1631	0.5778	0.1770	0.0900	0.1552	0.5778	0.1770	0.0900	0.1552	0.5990	0.1665	0.0875	0.1471	0.5990	0.1665	0.0875	0.1471	0.9919	0.0024	0.0034	0.0023	0.9986	0.0006	0.0002	0.0006
2	0.5161	0.1878	0.1016	0.1945	0.5161	0.1878	0.1016	0.1945	0.5358	0.1793	0.0995	0.1854	0.5358	0.1793	0.0995	0.1854	0.5560	0.1708	0.0971	0.1762	0.5560	0.1708	0.0971	0.1762	0.9932	0.0020	0.0027	0.0021	0.9987	0.0006	0.0001	0.0006
3	0.5846	0.1625	0.0834	0.1695	0.5846	0.1625	0.0834	0.1695	0.6061	0.1532	0.0813	0.1593	0.6061	0.1532	0.0813	0.1593	0.6277	0.1440	0.0789	0.1494	0.6277	0.1440	0.0789	0.1494	1.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
4	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	0.3319	0.0905	0.4797	0.0980	0.7216	0.1059	0.0571	0.1154
5	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	0.0792	0.3131	0.5479	0.0599	0.0988	0.7493	0.0821	0.0697
6	0.1833	0.5154	0.2025	0.0987	0.1833	0.5154	0.2025	0.0987	0.1753	0.5352	0.1927	0.0968	0.1753	0.5352	0.1927	0.0968	0.1671	0.5555	0.1828	0.0946	0.1671	0.5555	0.1828	0.0946	0.0015	0.9934	0.0046	0.0005	0.0002	0.9995	0.0002	0.0001
7	0.1676	0.5741	0.1738	0.0845	0.1676	0.5741	0.1738	0.0845	0.1581	0.5958	0.1636	0.0825	0.1581	0.5958	0.1636	0.0825	0.1486	0.6177	0.1535	0.0801	0.1486	0.6177	0.1535	0.0801	0.0000	1.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000
8	0.1875	0.5276	0.1885	0.0963	0.1875	0.5276	0.1885	0.0963	0.1786	0.5473	0.1796	0.0944	0.1786	0.5473	0.1796	0.0944	0.1697	0.5674	0.1707	0.0922	0.1697	0.5674	0.1707	0.0922	0.0032	0.9874	0.0084	0.0010	0.0010	0.9979	0.0009	0.0003
9	0.0905	0.1460	0.5856	0.1780	0.0905	0.1460	0.5856	0.1780	0.0876	0.1382	0.6076	0.1666	0.0876	0.1382	0.6076	0.1666	0.0845	0.1303	0.6297	0.1554	0.0845	0.1303	0.6297	0.1554	0.0177	0.0482	0.8570	0.0771	**0.0000**	**0.0000**	**1.0000**	**0.0000**
10	0.0923	0.1611	0.5827	0.1639	0.0923	0.1611	0.5827	0.1639	0.0894	0.1518	0.6043	0.1545	0.0894	0.1518	0.6043	0.1545	0.0862	0.1426	0.6262	0.1451	0.0862	0.1426	0.6262	0.1451	0.0172	0.0589	0.8640	0.0600	0.0002	0.0006	0.9986	0.0006
11	0.0963	0.1523	0.5648	0.1865	0.0963	0.1523	0.5648	0.1865	0.0934	0.1447	0.5868	0.1752	0.0934	0.1447	0.5868	0.1752	0.0902	0.1369	0.6089	0.1640	0.0902	0.1369	0.6089	0.1640	0.0180	0.0519	0.8466	0.0835	0.0001	0.0002	0.9994	0.0003
12	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	0.0669	0.1211	0.6831	0.1289
13	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	0.0756	0.0506	0.5051	0.3687	0.0731	0.0451	0.0682	0.8135
14	0.1381	0.0710	0.1385	0.6524	0.1381	0.0710	0.1385	0.6524	0.1271	0.0680	0.1274	0.6774	0.1271	0.0680	0.1274	0.6774	0.1165	0.0647	0.1167	0.7021	0.1165	0.0647	0.1167	0.7021	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**
15	0.1575	0.0867	0.1411	0.6147	0.1575	0.0867	0.1411	0.6147	0.1459	0.0830	0.1317	0.6394	0.1459	0.0830	0.1317	0.6394	0.1347	0.0791	0.1224	0.6639	0.1347	0.0791	0.1224	0.6639	0.0017	0.0005	0.0045	0.9933	0.0002	0.0001	0.0002	0.9995
16	0.1534	0.0854	0.1660	0.5952	0.1534	0.0854	0.1660	0.5952	0.1437	0.0824	0.1547	0.6191	0.1437	0.0824	0.1547	0.6191	0.1341	0.0791	0.1437	0.6431	0.1341	0.0791	0.1437	0.6431	0.0014	0.0004	0.0040	0.9942	0.0002	0.0001	0.0002	0.9996
	Data configuration 3)																															
cluster	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	0.3476	0.2443	0.1817	0.2264	0.3476	0.2443	0.1817	0.2264	0.3627	0.2375	0.1787	0.2210	0.3627	0.2375	0.1787	0.2210	0.3781	0.2308	0.1756	0.2155	0.3781	0.2308	0.1756	0.2155	0.3938	0.2240	0.1723	0.2099	0.3938	0.2240	0.1723	0.2099
2	0.3343	0.2382	0.1866	0.2408	0.3343	0.2382	0.1866	0.2408	0.3475	0.2331	0.1840	0.2354	0.3475	0.2331	0.1840	0.2354	0.3610	0.2279	0.1812	0.2299	0.3610	0.2279	0.1812	0.2299	0.3750	0.2225	0.1783	0.2243	0.3750	0.2225	0.1783	0.2243
3	0.3579	0.2368	0.1743	0.2310	0.3579	0.2368	0.1743	0.2310	0.3745	0.2298	0.1714	0.2244	0.3745	0.2298	0.1714	0.2244	0.3912	0.2228	0.1683	0.2178	0.3912	0.2228	0.1683	0.2178	0.4081	0.2158	0.1650	0.2111	0.4081	0.2158	0.1650	0.2111
4	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**
5	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**
6	0.2361	0.3240	0.2578	0.1821	0.2361	0.3240	0.2578	0.1821	0.2312	0.3375	0.2515	0.1799	0.2312	0.3375	0.2515	0.1799	0.2261	0.3513	0.2451	0.1775	0.2261	0.3513	0.2451	0.1775	0.2209	0.3656	0.2387	0.1749	0.2209	0.3656	0.2387	0.1749
7	0.2321	0.3441	0.2478	0.1760	0.2321	0.3441	0.2478	0.1760	0.2256	0.3610	0.2401	0.1732	0.2256	0.3610	0.2401	0.1732	0.2192	0.3780	0.2326	0.1702	0.2192	0.3780	0.2326	0.1702	0.2127	0.3951	0.2251	0.1670	0.2127	0.3951	0.2251	0.1670
8	0.2382	0.3419	0.2366	0.1833	0.2382	0.3419	0.2366	0.1833	0.2327	0.3552	0.2315	0.1807	0.2327	0.3552	0.2315	0.1807	0.2271	0.3688	0.2262	0.1779	0.2271	0.3688	0.2262	0.1779	0.2215	0.3828	0.2207	0.1750	0.2215	0.3828	0.2207	0.1750
9	0.1862	0.2135	0.3628	0.2375	0.1862	0.2135	0.3628	0.2375	0.1825	0.2081	0.3790	0.2304	0.1825	0.2081	0.3790	0.2304	0.1786	0.2026	0.3955	0.2233	0.1786	0.2026	0.3955	0.2233	0.1746	0.1969	0.4123	0.2162	0.1746	0.1969	0.4123	0.2162
10	0.1886	0.2199	0.3613	0.2303	0.1886	0.2199	0.3613	0.2303	0.1848	0.2141	0.3771	0.2240	0.1848	0.2141	0.3771	0.2240	0.1809	0.2082	0.3933	0.2176	0.1809	0.2082	0.3933	0.2176	0.1768	0.2022	0.4099	0.2111	0.1768	0.2022	0.4099	0.2111
11	0.1916	0.2150	0.3528	0.2406	0.1916	0.2150	0.3528	0.2406	0.1879	0.2100	0.3683	0.2338	0.1879	0.2100	0.3683	0.2338	0.1841	0.2048	0.3841	0.2270	0.1841	0.2048	0.3841	0.2270	0.1800	0.1995	0.4003	0.2202	0.1800	0.1995	0.4003	0.2202
12	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**
13	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**	**0.0000**	**0.0000**	**0.0000**	**1.0000**
14	0.2177	0.1674	0.2181	0.3968	0.2177	0.1674	0.2181	0.3968	0.2097	0.1635	0.2100	0.4168	0.2097	0.1635	0.2100	0.4168	0.2017	0.1592	0.2019	0.4372	0.2017	0.1592	0.2019	0.4372	0.1936	0.1547	0.1937	0.4580	0.1936	0.1547	0.1937	0.4580
15	0.2268	0.1878	0.2082	0.3773	0.2268	0.1878	0.2082	0.3773	0.2192	0.1830	0.2022	0.3956	0.2192	0.1830	0.2022	0.3956	0.2115	0.1780	0.1961	0.4144	0.2115	0.1780	0.1961	0.4144	0.2038	0.1728	0.1897	0.4338	0.2038	0.1728	0.1897	0.4338
16	0.2230	0.1758	0.2276	0.3736	0.2230	0.1758	0.2276	0.3736	0.2165	0.1724	0.2206	0.3904	0.2165	0.1724	0.2206	0.3904	0.2099	0.1687	0.2136	0.4079	0.2099	0.1687	0.2136	0.4079	0.2030	0.1647	0.2063	0.4259	0.2030	0.1647	0.2063	0.4259

Note: Medoids' membership degrees are in bold.

Table A.11: Membership degrees for simulation study 2, according to different combinations of β_1 and β_2, and to data configurations 3) and 4)

Data configuration	8, 0	(0, 8)	(12, 0)	(0, 12)	(16, 0)	(0, 16)	(20, 0)	(0, 20)
	(β_1, β_2)							
1)	0.90	0.89	0.90	0.89	0.90	0.89	0.90	0.89
2)	0.84	0.82	0.84	0.82	0.84	0.82	0.84	0.82
3)	0.56	0.60	0.57	0.57	0.58	0.58	0.84	0.75
4)	0.16	0.16	0.23	0.23	0.26	0.26	0.29	0.29

Table A.12: Fuzzy Silhouette index values for simulation study 2 according to different setting of the parameters β_1, β_2 (column wise) and to data configuration (row wise)