




Similarity Measures for Spatial Clustering

Leila Hamdad^(✉) , Karima Benatchba, Soraya Ifrez, and Yasmine Mohguen

Ecole nationale Supérieure en Informatique ESI,
BP 68M, 16309 Oued-Smar, Alger, Algeria
l.hamdad@esi.dz
<http://www.esi.dz>

Abstract. The spatial data mining (SDM) is a process that extracts knowledge from large volumes of spatial data. It takes into account the spatial relationships between the data. To integrate these relations in the mining process, SDM uses two main approaches: Static approach that integrates spatial relationships in a preprocessing phase, and dynamic approach that takes into consideration the spatial relationship during the process. In this work, we are interested in this last approach. Our proposition consists on taking into consideration the spatial component in the similarity measure. We propose two similarity measures; d_{Dyn1} , d_{Dyn2} . We will use those distances on the main task of SDM, spatial clustering, particularly on K-means algorithm. Moreover, a comparison between these two approaches and other methods of clustering will be given. The tests are conducted on Boston dataset with 590 objects.

Keywords: Spatial data mining · Dynamic approach
Similarity · Preprocessing approach · Clustering · K-means · DBSCAN

1 Introduction

These last decades have seen an explosion in the volume of spatial data. This is due to the various technological advances and the development of automatic data acquisition tools (GPS, satellite images,...). The wide use of these data has given rise to spatial data mining (SDM). It is a process that allows extracting knowledge and useful patterns from large volumes of spatial datasets [11, 15]. Spatial data consist of two types of components; a descriptive component containing data of usual type: integer, real, boolean,... describing some features of the data and geometric component which describes the spatial localization of the data. The SDM is more difficult than classical data mining as it handles geo-spatial data characterized by autocorrelation. In SDM, spatial component can be taken into account according to three different approaches. The first one is basic and considers all attributes in the same way (spatial and non-spatial) and uses the techniques of classical data mining. As a result, the auto-correlation between objects is not taken into consideration. In fact, several studies have shown that

this process produces inconsistent results, [14,19]. The second approach (static), likewise, uses classical DM techniques to process geo-spatial data. However, a preliminary data preprocessing is required. It consists of extracting and representing the geo-spatial relationship between entities, in an explicit manner [4,17]. This approach has many drawbacks. First of all, it is time consuming as the preprocessing of the data takes time. Indeed, extraction may require a lot of computation time when many relationships must be considered. Second, the preprocessed and the original data have to be stored leading to a redundancy in storage. Moreover, if the original data is modified, the pre-processing has to be executed again. To overcome these drawbacks, the dynamic approach emerged. It consists on taking into consideration the geo-spatial component during the data-mining process. Our proposal lies in this last approach, as, in the literature, spatial dynamic processing is not explicitly defined (see [12]). Our approach consists on taking into consideration the spatial component in the similarity measure. We will be interested, in this work, in the main task of SDM, clustering. Formally, it consists on partitioning a set S of N objects, $S = \{O_1, O_2, \dots, O_N\}$, based on a similarity metric, into a number of clusters (C_1, C_2, \dots, C_K) , such as: $C_i \neq \emptyset$ for $i = 1, \dots, K$, $C_i \cap C_j = \emptyset$ for $1, \dots, k$ and $j = 1, \dots, K$, $i \neq j$, $S = \cup_{i=1}^k C_i$. Objects of a cluster must be as similar as possible, while objects of different clusters must be as dissimilar as possible. It is an important task as it enables to show interesting objects grouping without a priori knowledge.

The objective of this paper is to propose two similarity distances that take into consideration the spatial component during the clustering process. To show the effectiveness of this proposal, we used these distances in K-means, a simple partitioning method yet efficient and widely used. We compared the obtained results to the first approach which we called static using K-means. Then we compare the three versions of K-means to density based clustering method DBSCAN [8] and CAH an ascending hierarchical clustering method.

This paper is organized as follows: In Sect. 2, we will introduce the static approach. Then in Sect. 3, we will present the similarity distances to be used for dynamic clustering. In Sect. 4, we present some clustering algorithms. Finally in Sect. 5, tests and results on the different approaches will be presented.

2 Static Approach

This approach allows taking advantage of traditional data mining methods to process spatial data. There are different types of spatial relationships: metric, topological and directional. In this paper, we deal with metric relations as the data considered are represented by points. The data preprocessing begins by extracting the metric relationships between the data. These relationships are then used to modify the attributes of data. There exist in the literature several methods to extract it. Moran [16] and Geary [9] were the first to propose a measure for spatial interaction. Since then, several measures have been proposed [3]. Most of them require a threshold as parameter (obtained from the objects' distance matrix) [13]. We have chosen to use the K nearest neighbors (KNN) as

it is a simple non parametric method (does not need a threshold); moreover, it is the most used method [13]. KNN processes as follows: For a given object O_i of the dataset, KNN searches among all objects of the data set, the K-nearest neighbors of O_i according to a distance such as the geometric distance. The neighbor relations obtained by applying KNN on the original dataset are used to obtain neighborhood matrix. The last step of the static approach consists on building the new dataset (smoothing matrix) from the original one. In fact, each attribute value of an object is replaced by the mean of its neighbors attribute. The algorithm is given below:

Smoothing Matrix algorithm For each object O_i of S , $i = 1, \dots, N$:

- (1) Compute K-nearest- neighbors of O_i with KNN algorithm.
- (2) For each attribute of O_i , compute the mean of the attributes of O_i and its corresponding neighbors' attribute.
- (3) Create a new object O'_i which attributes values are the one computed in (2).
- (4) Insert O'_i in S' where S' is the new dataset.

3 Similarity Distances for a Dynamic Approach

This approach, unlike the previous one, proposes dynamic spatial data processing. Its goal is to take into account the spatial component in the process of DM and not through a preprocessing. However, in the literature very little work is available on this approach [12]. This might be due to the fact that this research area is relatively young [17]. One of the goals of this paper is to take into consideration the spatial component, in a similarity distance. We propose two distances d_{Dym1} and d_{Dym2} . Both compute the similarity between spatial objects by considering the non-spatial attributes and spatial attributes simultaneously. They are based on two distances: Euclidean distance that measures the similarity between the non-spatial attributes and geographic distance that measures the metric relationship. The first one, d_{Dym1} is formally defined as the product of the Euclidean distance between the non-spatial attributes and geographic distance between the spatial attributes:

$$d_{Dym1}(O_i, O_k) = d_{euclidian}(O_i, O_k) * d_{geo}(O_i, O_k).$$

Where

$$d_{euclidian}(O_i, O_k) = \sqrt{\sum_{j=1}^m (O_i - O_k)^2}$$

and

$$d_{geo}(O_i, O_k) = 6371 * \text{Acos}[\text{Cos}(lat1) * \text{Cos}(lat2) * \text{Cos}(long2 - long1) + \text{Sin}(lat1) * \text{Sin}(lat.2)].$$

Where $(lat1, long1)$ and $(lat2, long2)$ are the spatial coordinates of respectively points O_i and O_k . In the second proposed distance d_{Dym2} , weights are assigned

to the two types of attributes (spatial and non-spatial). The formula of $d_{D_{yn}2}$ is given below:

$$d_{D_{yn}2}(O_i, O_k) = \alpha d_{euclidian}(O_i, O_k) + \beta d_{geo}(O_i, O_k).$$

where α and β are respectively weights of Euclidian and geographic distance and verify that $\alpha + \beta = 1$. When $\alpha = 1(\beta = 0)$, clustering is done on descriptive attributes only. The spatial attributes are ignored. On the contrary, when $\alpha = 0(\beta = 1)$, only spatial attributes are considered and we obtain a regionalization.

4 Clustering Algorithms

Many methods dedicated to the clustering exist in the literature. They fall into two main classes: partitioning methods and hierarchical methods. They differ in the way they build clusters. While the second gradually build those clusters, the first discover them by moving objects between clusters [11]. In addition to these two fundamental approaches, other methods exist such as density and grid based methods. They use different mechanisms for data organization and processing, and for building the clusters [1].

4.1 Partitioning Methods

They seek to find the best k partitions for a set of n objects (data), while optimizing an objective function. This function aims to maximize the similarity between objects of the same cluster and to minimize the similarity between objects of different clusters. These methods improve iteratively clusters by moving objects between clusters. There exist several partitioning methods among which K-means, the first clustering method. K-means is by far the most popular clustering algorithm and widely used in scientific and industrial applications [5]. Its popularity is due to the fact that it is simple to implement and converges rapidly. However, it has some drawbacks. It is influenced by outliers and the obtained clustering depends on the initial one.

4.2 Hierarchical Methods

These clustering methods build hierarchical clusters gradually. They can be of two types ascending hierarchical clustering methods and descending ones. Hierarchical methods have many advantages such as flexibility regarding the level of granularity one wants to have. However they present some drawbacks such as the difficulty of setting a stopping criterion and their high execution time [6]. In what follows, we present, CAH, an ascending hierarchical clustering method. CAH builds a hierarchy of clusters assuming that initially each object is a cluster. Then the most similar clusters are aggregated. The aggregation is repeated until having a single cluster containing all objects.

4.3 Density Based Methods

These methods characterize classes as homogeneous high density regions, separated by regions of low density. Unlike partitioning methods, methods based on density are able to discover classes of concave forms and do not take into account outliers as they are removed during the process. There are two approaches for this type of methods. The first approach is based on the density connectivity. It takes two input parameters, the neighborhood radius Eps which represents the maximum distance between points and the density threshold $MinPts$ which represents the minimum number of points in the neighborhood. The best known algorithms are DBSCAN and GDBSCAN [2]. The second approach is based on sound mathematical principles and uses a density function in its process. The best known algorithm is DENCLUE [10].

5 Tests and Results

In this section, we will give a summary of the different tests performed. We will start by presenting the result of k-means using the two proposed distances d_{Dyn1} and d_{Dyn2} , respectively named $K\text{-means}_{d_{Dyn1}}$ and $K\text{-means}_{d_{Dyn2}}$. K-means with preprocessing using KNN is called $K\text{-means}_{static}$. Then we will compare those results to DBSCAN and CAH algorithms. These tests were executed on an unsupervised spatial benchmark Boston¹, represented in Fig. 1. Benchmark Boston contains 506 objects and represents Boston housing and neighborhood data. Each object is represented by 18 non spatial features and longitude and latitude (spatial attributes). The different features describe the type of area of the different housing such as: hot district characterized by high criminality level, rich district, poor one... One can notice that this benchmark is very dense.

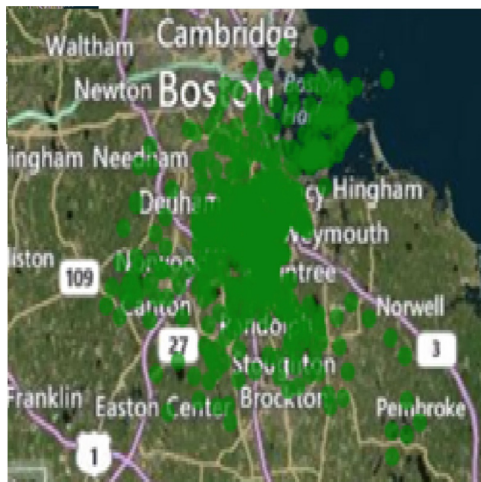


Fig. 1. Representation of benchmark Boston.

¹ <https://geodacenter.asu.edu>.

5.1 Dynamic vs Static K-means

To compare the dynamic versions of K-means ($K\text{-means}_{d_{Dyn1}}$ and $K\text{-means}_{d_{Dyn2}}$) to the static one we conducted several tests. For all the tested version of K-means, the number of iterations was set to 100. As this benchmark is unsupervised one, and to compare the results of the three versions of k-means, we executed classical k-means ($k = 3$) on the non spatial attributes of the benchmark to better visualize the objects distribution and their meaning. Moreover, we executed $K\text{-means}_{d_{Dyn2}}$ on spatial attributes ($\alpha = 0$), obtaining a regionalization. The results of both executions are given in the following Figures. In Fig. 2, the blue color represents north districts, the yellow one the center, and the red one south districts. In Fig. 3, the blue color represents the wealthy neighborhood, the yellow one the middle class neighborhood and the red color the poor neighborhood.

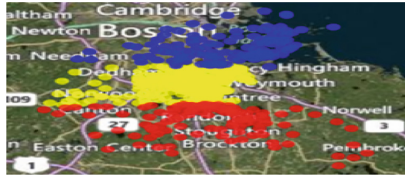


Fig. 2. Regionalization. (Color figure online)

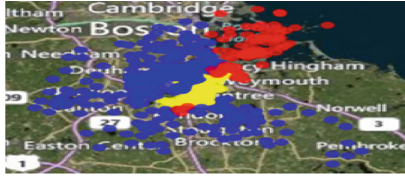


Fig. 3. Classical K-means. (Color figure online)

5.2 Comparison of $K - means_{static}$, $K - means_{d_{Dyn1}}$ and $K - means_{d_{Dyn2}}$

Figures 4, 5, 6 and 7 show the results of visualization of $K\text{-means}_{static}$ (with the number of neighbors for KNN being to 2 and 4), $K\text{-means}_{d_{Dyn1}}$ and $K\text{-means}_{d_{Dyn2}}$ ($\alpha = \beta = 0.5$). For these tests, the number of classes for K-means was set to 3.

If we take a closer look at Figs. 4 and 5, we can see the influence of the KNN parameter on the clustering. For a *number of neighbors* = 2, a certain number of objects (the ones between the red class and the yellow one) are considered belonging to the cluster “poor neighborhood”. While when more neighbors are taken into consideration when building the smoothing matrix, this number decreases. One notes that some objects classified in the blue cluster in Fig. 4 are classified in the red cluster in Fig. 5.

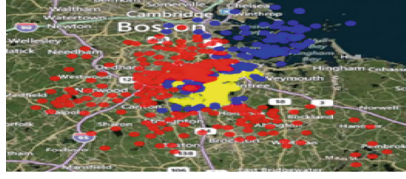


Fig. 4. Static K-means with number of neighbors = 2. (Color figure online)

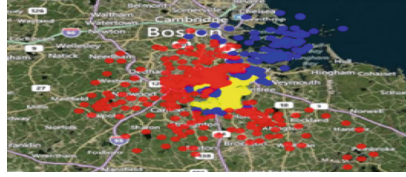


Fig. 5. Static K-means with number of neighbors = 4. (Color figure online)



Fig. 6. K-means Dyn1. (Color figure online)

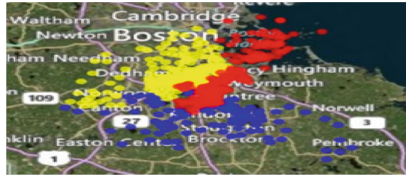


Fig. 7. K-means Dyn2 ($\alpha = \beta = 0.5$). (Color figure online)

If we compare the results of classical K-means and those of $K\text{-means}_{static}$ (Fig. 5), we notice that the two classifications can be considered as close. We do not see a clear impact of the spatial attributes on the clustering. One notices that the boundaries between classes are clearly defined for $K\text{-means}_{d_{Dyn1}}$ (Fig. 6) when compared to $K\text{-means}_{static}$ (Figs. 4 and 7). Classes do not overlap as for $K\text{-means}_{static}$. This may be due to the distance used which gives the same importance to both types of attributes. $K\text{-means}_{d_{Dyn2}}$ gives a different partitioning of the objects. A close look at Fig. 7 shows that two classes of $K\text{-means}_{d_{Dyn1}}$ (the poor neighborhood in red and the middle class neighborhood in yellow, Fig. 6) are merged to form the poor neighborhood (in red). While the rich neighborhood of Fig. 6 is split into two clusters: yellow one for middle class neighborhood

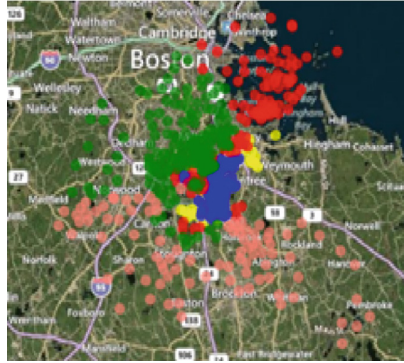


Fig. 8. Results' visualization of K-means static (with 4 neighbors and $K = 5$). (Color figure online)

and blue one for rich neighborhood. To try to explain this, we executed the $K\text{-means}_{static}$ with 4 neighbors for KNN, setting the number of classes to 5. We obtained the following results (Fig. 8): The obtained classes of this execution are described as follows: The pink cluster represents the rich neighborhood, the green one the middle class one, the yellow one the residential area, the blue the economical one and the red one the poor neighborhood. Given this type of information, we can conclude that the clustering of $K\text{-means}_{dDyn2}$ is interesting as it merged the economical and poor neighborhood, found by $K\text{-means}_{static}$ and $K\text{-means}_{dDyn1}$, into one class (Red one in Fig. 7). It also, separated the rich class found by $K\text{-means}_{static}$ and $K\text{-means}_{dDyn1}$ into two classes: Middle class and rich neighborhood. Now, if we compare the different approaches according to the intra cluster inertia and execution time (Figs. 9 and 10), we note that as the number of classes increases, the inertia decreases. This is due to the fact that as the size of clusters decreases, clusters will have more similar objects. For 3 clusters, the best inertia is given by $K\text{-means}_{static}$ with 2 neighbors, followed by $K\text{-means}_{dDyn2}$. However, the dynamic version of K-means is faster than the preprocessing k-means ($K\text{-means}_{static}$). That was the goal of using dynamic K-means.

5.3 Dynamic K-means vs DBSCAN and CAH

In this section, we will compare $K\text{-means}_{dDyn1}$ and $K\text{-means}_{dDyn2}$ to DBSCAN and CAH. Figures 11, 12, 13 and 14 display the results of respectively $K\text{-means}_{dDyn1}$, $K\text{-means}_{dDyn2}$, DBSCAN and CAH.

We can notice from these Figures, that the four methods give different clustering. DBSCAN (Fig. 13) give more compact classes. One cluster (rich neighborhood in blue) is bigger, in terms of size than the other two. $K\text{-means}_{dDyn2}$ divide this class into two (rich and middle classe neighborhood). CAH (Fig. 14) does the same; however, its rich neighborhood is smaller and is located south. Both $K\text{-means}_{dDyn2}$ and CAH define the same poor neighborhood.

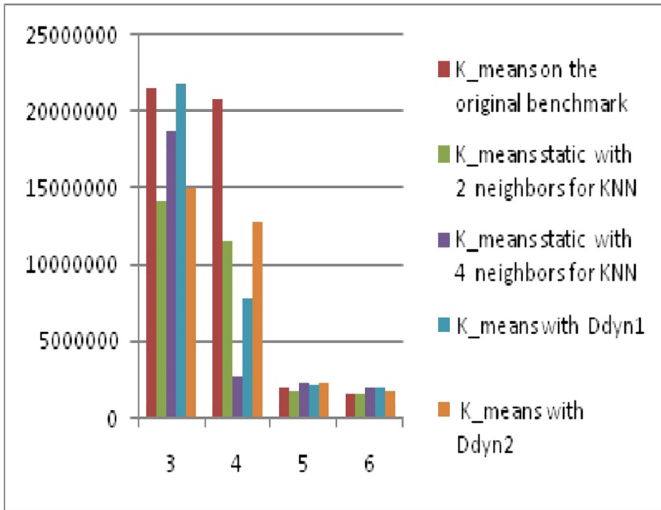


Fig. 9. Intra cluster inertia of the different k-means versions when varying k.

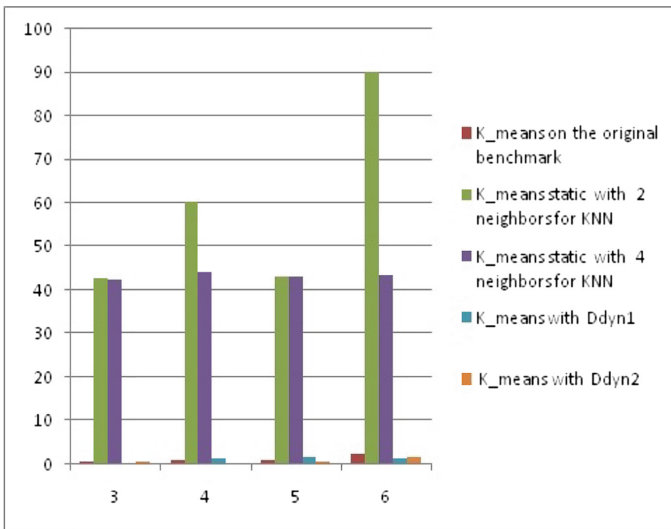


Fig. 10. Execution time of the different versions of K-means when varying K.



Fig. 11. $K - means_{Ddyn1}$.

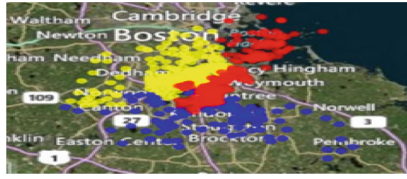


Fig. 12. $K\text{-means}_{d_{Dyn2}}(\alpha = \beta = 0.5)$.



Fig. 13. DBSCAN ($Eps = 100$ et $MinPts = 5$).



Fig. 14. CAH ($\alpha = \beta = 0.5$).

If we compare the intra cluster inertia of DBSCAN, CAH and $K\text{-means}_{d_{Dyn2}}$ (see Fig. 15), we notice that, DBSCAN has the best followed by $K\text{-means}_{d_{Dyn2}}$ and CAH. However, for this method, objects that are distant or isolated are considered noises and are not assigned to clusters. In Fig. 13, these points are in black and represent 3.4% of the size of the benchmark. While the execution time of DBSCAN and $K\text{-means}_{d_{Dyn2}}$ somehow similar, CAH takes much more time (Fig. 15).

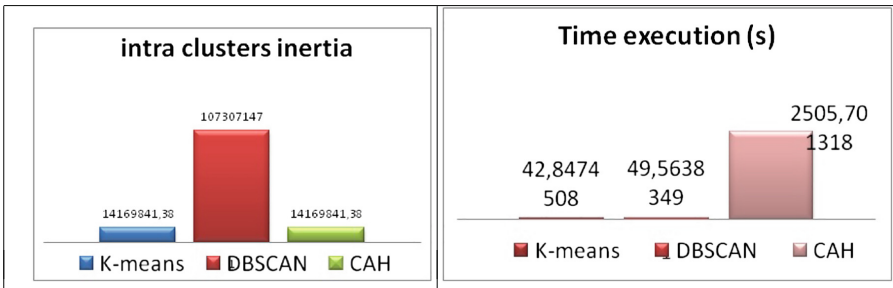


Fig. 15. Comparison between DBSCAN, CAH and $K\text{-means}_{d_{Dyn2}}$.

6 Conclusions

We focused, in our work, on the main descriptive task of SDM, spatial clustering. The main goal of our study was to compare two spatial clustering approaches namely: the spatial data preprocessing approach (static) and dynamic approach. For the second approach, we proposed to take into consideration the spatial component in the similarity measure. We proposed two distances, $d_{D_{yn1}}$ and $d_{D_{yn2}}$. The various tests performed on the benchmark Boston, showed that the approach proposed ($K - means_{d_{D_{yn1}}}$, $K - means_{d_{D_{yn2}}}$) gives better results than the preprocessing approach in terms of execution time. This was the main goal as the preprocessing approach takes too much time. The results obtained with $K - means_{d_{D_{yn1}}}$ seems similar to those of the preprocessing approach but with more precise boundaries. This is due to the fact that $K - means_{d_{D_{yn1}}}$ takes into consideration both spatial and non spatial attributes. $K - means_{d_{D_{yn2}}}$ is more efficient in terms of intra-class inertia, it makes a good description in terms of regionalization and characterization of these obtained regions.

References

1. Andritsos, P.: Data clustering techniques. Technical report CSRG-443, University of Toronto (2002)
2. Ankerst, M., Breunig, M., Kriegel, H., Sander, J.: OPTICS: ordering points to identify the clustering structure. In: ACM SIGMOD, International Conference on Management of Data (1999)
3. Anselin, L.: Spatial Econometrics: Methods and Models. Springer, Dordrecht (1988). <https://doi.org/10.1007/978-94-015-7799-1>
4. Bogorny, V., Martins, E., Alvares, L.O.: A reuse-based spatial data preparation framework for data mining. In: Proceedings of the 17th International Conference on Software Engineering and Knowledge Engineering, pp. 649–652 (2005)
5. Boubou, M.: Contribution aux méthodes de classification non supervisée via des approches prétopologiques et d'aggrégation d'opinions. Thèse de doctorat, Université Claude Bernard - Lyon I (2007)
6. Cleuziou, G.: Une méthode de classification non-supervisée pour l'apprentissage de règles et la recherche d'information. Thèse de doctorat, Université d'Orléans (2004)
7. Ester, M., Kriegel, H.-P., Sander, J.: Spatial data mining: a database approach. In: Scholl, M., Voisard, A. (eds.) SSD 1997. LNCS, vol. 1262, pp. 47–66. Springer, Heidelberg (1997). https://doi.org/10.1007/3-540-63238-7_24
8. Ester, M., Kriegel, H.P., Sander, J.: Knowledge discovery in spatial databases. In: Institute for Computer Science, University of Munich, Invited Paper at 23rd German Conference on Artificial Intelligence (KI 1999), Bonn, Germany (1999)
9. Geary, R.C.: The contiguity ratio and statistical mapping. *Incorporated Stat.* **5**(3), 115–145 (1954)
10. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Elsevier Inc. (2006)
11. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: a review. *ACM Comput. Surv.* **31**(3), 264–323 (1999)
12. Klosgen, W., Zytkow, J.M.: Knowledge discovery in databases: the purpose, necessity, and challenges. In: Handbook of Data Mining and Knowledge Discovery. Oxford University Press, Inc. L.Y (2002)

13. Lebart, L.: Contiguity analysis and classification. In: Gaul, W., Opitz, O., Schader, M. (eds.) *Data Analysis. Studies in Classification, Data Analysis, and Knowledge Organization*, pp. 233–243. Springer, Heidelberg (2000). https://doi.org/10.1007/978-3-642-58250-9_19
14. Maguire, D.J.: An overview and definition of GIS. In: *Geographic Information Systems: Principles and Applications*, 1st edn., vols. 1, 2, pp. 9–20 (1991)
15. Mennis, J., Guo, D.: Spatial data mining and geographic knowledge discovery an introduction. *Comput. Environ. Urban Syst.* **33**(6), 403–408 (2009)
16. Moran, P.A.P.: The interpretation of statistical maps. *Biometrika* **35**, 255–260 (1948)
17. Rinzivillo, S., Turini, F., Bogorny, V., Körner, C., Kuijpers, B., May, M.: Knowledge discovery from geographical data. In: Giannotti, F., Pedreschi, D. (eds.) *Mobility, Data Mining and Privacy*, pp. 243–265. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-75177-9_10
18. Tobler, W.R.: Cellular geography. In: Gale, S., Olsson, G. (eds.) *Phylosophy in Geography*, pp. 379–386. Reidel, Dortrecht (1979)
19. Zeitouni, K., Yeh, L.: Le data mining spatial et les bases de données spatiales. *Revue internationale de géomatique, Numéro spécial sur le Data mining spatial* **9**(4) (2000)