CLUSTERING USING GRAPH THEORY TOOLS

Jakub Danko – Tomáš Löster

Abstract

The aim of this paper is to introduce and describe a method for the classification of objects into a predetermined number of groups using the discrete mathematical tool – graph theory. This clustering method is based on the most commonly used Euclidean distance, from which a complete graph is then constructed. From this complete graph, the minimum spanning tree of the graph is estimated. We have chosen this method because it is known that the minimum spanning tree can find certain structures in the data and therefore we assume that it also has a classification potential. Through this spanning tree, we seek to assign objects to groups that minimize the overall distance in the graph. The results of the assignment are then compared with the available hierarchical clustering methods, which are also based on the Euclidean distance for correctness. We present the results obtained by applying the method to the Iris dataset.

Key words: clustering, graph theory, Iris, minimum spanning tree

JEL Code: C38, C61

Introduction

Cluster analysis is a mathematical-statistical method whose primary objective is to classify objects into groups called clusters. An object is an observation that is characterized by various variables. These variables may generally be quantitative or qualitative, or combinations thereof. The basic aim of cluster analysis is to make objects belonging to one cluster as similar as possible and objects belonging to two different clusters to be as dissimilar as possible. There are many clustering methods in the current literature that can be divided according to different criteria. The development of cluster analysis methods is associated with the appearance of new methods and modifications of existing ones. We present a new clustering method using graph theory tools, namely the minimum spanning tree method for solving the classification problem on the well-known dataset *Iris*. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class 2 (Iris setosa, see Fig. 1) is linearly separable from the other 2. The latter are not linearly separable from each other. Attributes are in centimeters and represent sepal length, sepal width, petal length, and petal width.

1 Cluster analysis

The term *"cluster analysis"* was first used by R. C. Tryon, who understood it as an objective procedure to group objects based on their similarity and diversity. (Tryon, 1939) This is a method that investigates the similarities of multidimensional objects. The analyzed data matrix is of dimension *n x p*, where *n* is the number of objects and *p* is the number of analyzed attributes of these objects (variables). The goal is to split a set of objects into several relatively homogeneous subsets (clusters) so that objects belonging to the same cluster are as similar as possible, while objects from different clusters should be as dissimilar as possible. Clustering methods are divided into two basic groups: hierarchical and non-hierarchical clustering methods. The difference between these two methods is that hierarchical searches for additional clusters using already created clusters, while non-hierarchical methods identify all clusters at once. In our contribution, we propose our own clustering method using graph theory tools and compare the results of this assignment with the most commonly used hierarchical clustering methods, among which we consider Ward´s method, average linkage method, single linkage method, and complete linkage method of clustering. All clustering methods are based on some measure of distance or dissimilarity between units. In our case, we consider the so-called Euclidean distance, which is generally used most often in the case of quantitative continuous variables. Consider vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ representing p of the analyzed attributes of two objects of the data matrix. Then the Euclidean distance of vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ is given by the Formula 1:

$$D(\boldsymbol{x},\boldsymbol{y}) = \sqrt{\sum_{i=1}^{p}(\boldsymbol{x_i} - \boldsymbol{y_i})^2} \qquad (1)$$

Cluster analysis is widely used not only in economics but also in other scientific disciplines, for example in biology, medicine, etc. Rollnik-Sadowska and Dąbrowska (2018) used cluster analysis to compare the effectiveness of labor market policies among the European Union. Megyesiová and Lieskovská (2018) compared indicators of sustainable growth of OECD countries and one of the methods used for comparison was also cluster analysis. Miłek (2018) devoted himself to spatial differentiation in the social and economic development level in Poland using Ward's agglomerative clustering method. For an example of the use of cluster analysis in medicine, see Liao et al. (2016)

The analysis is performed using the statistical programming language **R**. Library *igraph* is used for working with graphs. (Csardi G. and Nepusz T., 2006)

2 Clustering using graph theory

For example, Hubert (1974) has devoted himself to graph theory in the context of cluster analysis. Ling and Killough (1976) focused on so-called random graphs and created probability tables for cluster analysis based on this approach. The algorithm which, like ours, uses the minimum spanning tree method used Zahn (1971) who demonstrated how the minimum spanning tree can be used to detect clusters. His algorithm consists of three steps:

1. construct the minimum spanning tree,
2. identify inconsistent edges in this tree,
3. remove the inconsistent edges to form connected components (clusters).

The most difficult part of the algorithm is to define the inconsistency. Zahn considers various criteria for inconsistency, for example, that edge is inconsistent if its interpoint distance is significantly larger than the average of nearby edge weights.

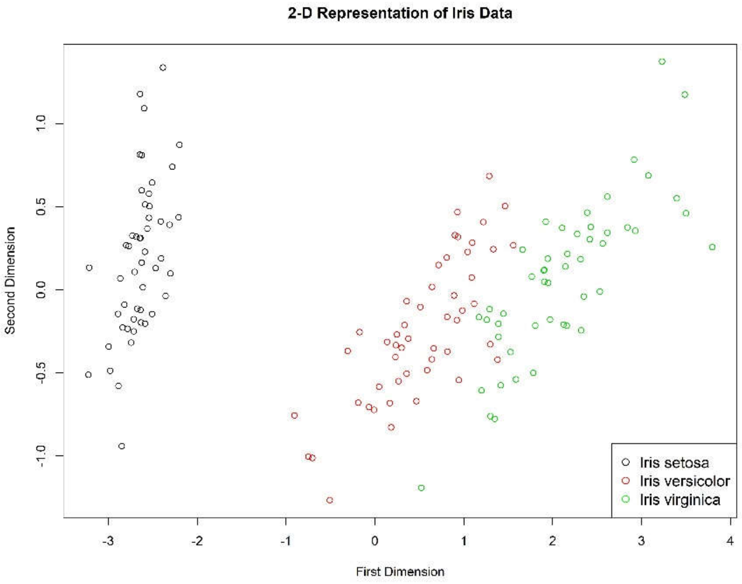
2.1 Proposal for new graph theory approach for clustering

Our algorithm is based on the theory of the minimum spanning tree. Since we compare the results with hierarchical methods that consider Euclidean distance, we first calculate the distance matrix using Formula 1. Since the dataset we analyze is of dimension *150 x 4* (*150* observations of three different types of irises, each is described by 4 quantitative variables), the resulting distance matrix is of dimension *150 x 150*. New mathematical objects, complete graphs, which are given by vertices and edges, are received from a distance matrix with the application of discrete mathematics tools. Vertices represent observations and the edges between the vertices represent distances between these observations from a distance matrix.

When applying the method of minimum spanning tree, which is finding such a subgraph of the original graph, that is continuous, does not contain cycles and has minimal edge evaluation considering that there is a path between every pair of vertices, we get so-called minimum spanning tree. This minimum spanning tree represents such a structure, in which objects that are the closest to each other are mutually linked.

Since we know that there are three kinds of irises, the goal is to create a classification where the number of clusters will be equal to three. First, let's visualize the 150 objects using a multidimensional scaling graph. (Eckart and Young, 1936) The goal of multidimensional scaling is to find the coordinates of points in the space of a given dimension so that the distances between objects in that space correspond as far as possible to the distances from the distance matrix. In our case, because of the visualization, we design a four-dimensional space into a two-dimensional space, so we are looking for a pair of coordinates for each object. Visualization is shown in Figure 1.

Fig. 1: Multidimensional scaling for the Iris dataset



Source: own processing using R

We can see that these are relatively well-separated objects. In particular, the species Iris setosa is located far from the other two species. These two species (Iris versicolor and Iris virginica) slightly overlap, so even in the case of cluster analysis we expect slightly ambiguous assignments. When applying the best-known hierarchical clustering methods using Euclidean distance, we get the following results listed in Tab. 1. (correct assignment is shown in bold, incorrect by italics)

Tab. 1: Results of hierarchical clustering using Euclidean distance

Complete linkage method	Iris setosa	Iris versicolor	Iris virginica
Iris setosa	**50**	0	0
Iris versicolor	0	**23**	*27*
Iris virginica	0	*49*	**1**
Ward´s method	Iris setosa	Iris versicolor	Iris virginica
Iris setosa	**50**	0	0
Iris versicolor	0	**49**	*1*
Iris virginica	0	*15*	**35**
Single linkage method	Iris setosa	Iris versicolor	Iris virginica
Iris setosa	**50**	0	0

Iris versicolor	0	**50**	0
Iris virginica	0	*48*	**2**
Average linkage method	Iris setosa	Iris versicolor	Iris virginica
Iris setosa	**50**	0	0
Iris versicolor	0	**50**	0
Iris virginica	0	*14*	**36**

Source: own processing using R

As can be seen in Tab. 1, the most successful classification method, in this case, is the Average linkage method, which managed to correctly assign all species of Iris setosa and Iris versicolor and 14 species of Iris virginica erroneously classified as Iris versicolor. This is followed by Ward´s method with 16 wrong assignments, the single linkage with 48 wrong assignments, and the worst classification method, in this case, is the complete linkage method with 76 wrong assignments, which represents more than 50% classification error rate. Furthermore, we can see that our assumption has been confirmed and all methods correctly identified in 100% of cases the species of Iris setosa that is well separable.

The clustering algorithm we propose consists of the following steps:

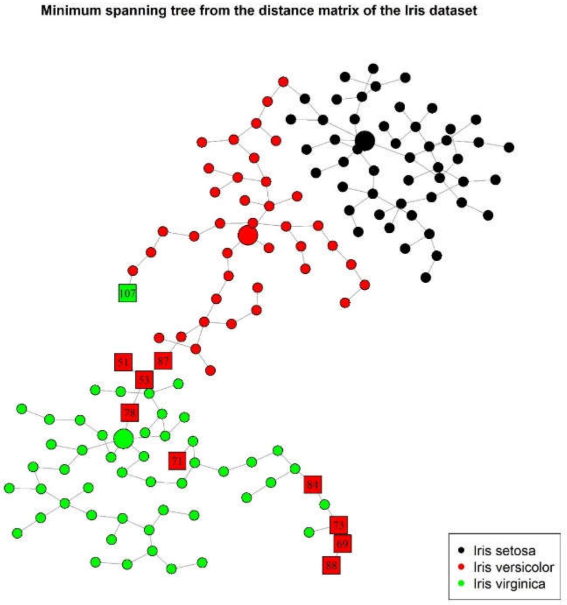
1. calculation of distances between observations using selected distance metric (we consider Euclidean distance in the contribution),
2. constructing a complete graph from a distance matrix,
3. constructing a minimum spanning tree from the complete graph,
4. generate random *k*-element subsets of vertices from the minimum spanning tree, where *k* is the number of clusters (in our case 3),
5. for each vertex, we calculate the distance to its nearest vertex from randomly generated *k* vertexes,
6. calculate the total distance of all vertices from the *k*-element subset of vertices,
7. select an assignment that minimizes this total distance.

Since the computational complexity of this algorithm is very high, for example, in the case of 150 objects we would have to create $\binom{150}{3} = 551300$ threes of vertices, then we would still recalculate their distance in the minimum spanning tree to these vertices for the remaining 147, we make the algorithm a little bit simpler. Obviously, the three vertices we're looking for will be somewhere in the *center* of a minimum spanning tree and not on its *edge*. Therefore, we narrow the selection of vertices only to vertices with a larger degree[1]. This way

[1] In graph theory, the degree of a vertex is the number of edges connecting it.

we can significantly reduce the computational time of the algorithm. If in our case we only consider vertices that have a degree greater than 3, of which there are 10 in our minimum spanning tree, we create only $\binom{10}{3} = 120$ threes of vertices which we consider a significant simplification and the results are exactly the same as the algorithm that considers all possible triplets. We present the results of our classification in Fig.2.

Fig. 2: Minimum spanning tree from the distance matrix of the Iris dataset

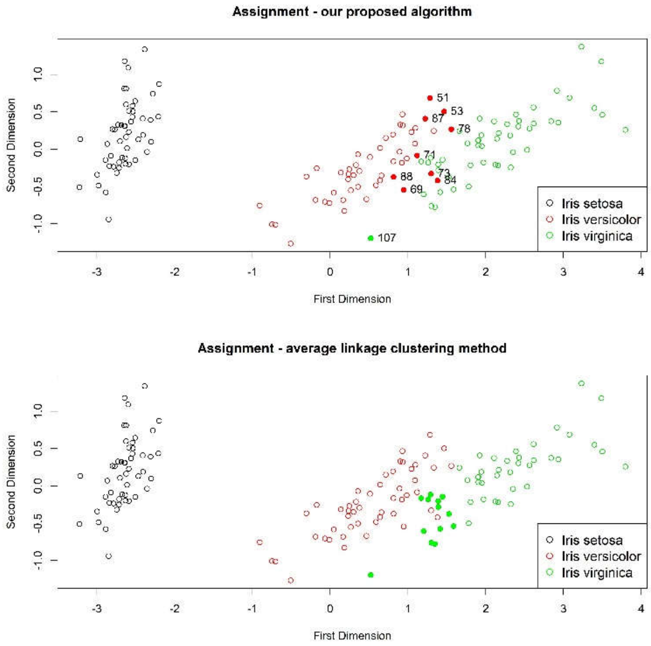


Source: own processing using R

As can be seen in Fig. 2, the actual membership of each group is indicated in color. The trinity of the vertices minimizing the total distance of the other vertices from them is displayed in large points. The incorrect assignment is displayed as a square in the graph. We can see that our algorithm incorrectly assigned one species of Iris virginica as Iris versicolor (107th observation) and 9 species of Iris virginica incorrectly assigned to the group Iris versicolor (observations 51, 53, 69, 71, 73, 78, 84, 87 and 88), which represents a classification error rate of 6.7%. For comparison, the best average linkage hierarchical clustering method has a classification error rate of 9.3%.

The method we proposed for this particular dataset was able to classify objects into groups better than any other hierarchical clustering method, using the same distance (considering Euclidean distance) to maintain the correctness of the comparison. Objects that have been incorrectly assigned are shown in Fig. 3, where we can see that they are really those objects that are on the border between Iris versicolor and Iris virginica. Interestingly, hierarchical algorithms have a problem with identifying a group of Iris virginica, which our algorithm can identify very well (1 misalignment) and, conversely, have no problem with identifying Iris versicolor, where our algorithm misaligned in 9 out of 50 cases. The incorrectly assigned objects are visualized on the graph by filled points.

Fig. 3: Comparison of the success of the assignment of our proposed algorithm and average linkage clustering method



Source: own processing using R

Conclusion

In this work, we dealt with the possibility of constructing a new clustering technique using graph theory. We have shown that on a particular dataset, the method we propose performs better than any hierarchical clustering method. Specifically, the best result for hierarchical clustering was using the average linkage hierarchical clustering method which has a classification error rate of 9.3%. The method we propose has for a dataset Iris classification error rate of 6.7%. The results confirm our assumption that the minimum spanning tree is a very good method for identifying structures and has classification possibilities. In the future, the analysis could continue with the analysis of other datasets and considering other types of distances, while the results of the classification will still be compared with existing clustering methods.

References

Csardi G, Nepusz T. (2006). The igraph software package for complex network research, InterJournal, Complex Systems 1695. http://igraph.org

Eckart C. & Young G. (1936). The approximation of one matrix by another of lower rank. Psychometrika. Volume 1. doi:10.1007/BF02288367

Jain A. K. & Dubes R. C. (1988). Algorithms for clustering data. Prentice-Hall, Inc. Upper Saddle River, NJ, USA. ISBN:0-13-022278-X

Hubert, L.J. (1974). Some applications of graph theory to clustering. Psychometrika 39: 283. https://doi.org/10.1007/BF02291704

Liao, M., Li, Y., Kianifard, F., Obi, E., & Arcona, S. (2016). Cluster analysis and its application to healthcare claims data: a study of end-stage renal disease patients who initiated hemodialysis. BMC nephrology, 17, 25. doi:10.1186/s12882-016-0238-2

Ling, R. F. & Killough G. G. (1976). Probability Tables for Cluster Analysis Based on a Theory of Random Graphs. Journal of the American Statistical Association Vol. 71, No. 354

Megyesiová, S. & Lieskovská V. (2018). Analysis of the Sustainable Development Indicators in the OECD Countries. In Sustainability.Basel: MDPI. ISSN 2071-1050, vol. 10, no. 12

Miłek, D. (2018). Spatial differentiation in the social and economic development level in Poland. Equilibrium. Quarterly Journal of Economics and Economic Policy, 13(3), 487–507. doi: 10.24136/eq.2018.024

Rollnik-Sadowska, E., & Dąbrowska, E. (2018). Cluster analysis of effectiveness of labour market policy in the European Union. Oeconomia Copernicana, 9(1), 143–158. doi: 10.24136/oc.2018.008

Tryon, R.C. (1939) Cluster Analysis: Correlation Profile and Orthometric (Factor) Analysis for the Isolation of Unities in Mind and Personality. Edwards Brothers, Ann Arbor

Zahn C. T. (1971). Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters. IEEE Transactions on Computers, vol. C-20, no. 1. doi: 10.1109/T-C.1971.223083

Contact

Jakub Danko
University of Economics, Prague
nám. W. Churchilla 4, 130 67 Prague,
Czech Republic
jakub.danko@vse.cz

Tomáš Löster
University of Economics, Prague
nám. W. Churchilla 4, 130 67 Prague,
Czech Republic
tomas.loster@vse.cz