Swarm Intelligence for Self-organized Clustering (Extended Abstract)*

Michael C. Thrun and **Alfred Ultsch**

Databionics Research Group, Philipps-University of Marburg, Germany

mthrun@informatik.uni-marburg.de, ultsch@informatik.uni-marburg.de

Abstract

The Databionic swarm (DBS) is a flexible and robust clustering framework that consists of three independent modules: swarm based projection, high-dimensional data visualization and representation guided clustering. The first module is the parameter-free projection method Pswarm, which exploits concepts of self-organization and emergence, game theory, and swarm intelligence. The second module is a parameter-free high-dimensional data visualization technique called topographic map. It uses the generalized U-matrix, which enables to estimate first, if any cluster tendency exists and second, the estimation of the number of clusters. The third module offers a clustering method which can be verified by the visualization and vice versa. Benchmarking w.r.t. conventional algorithms demonstrated that DBS can outperform them. Several applications showed that cluster structures provided by DBS are meaningful. Exemplary, a clustering of worldwide country-related data w.r.t. the COVID-19 pandemic is presented here. Code and data is made available via open source.

1 Introduction

The term knowledge discovery refers to the general process of finding valid, novel, potentially useful, and understandable patterns in data [Fayyad *et al.*, 1996]. Our focus lies on data-driven methods that find patterns in data that identify homogeneous groups of objects if these objects are heterogonous between the groups or so-called clusters [Bonner, 1964]. In this sense, cluster analysis can be seen as one step in the knowledge discovery process, and the clusters are often specified as "natural" clusters [Duda *et al.*, 2001; Theodoridis and Koutroumbas, 2009]. The question that arises is how to recognize structures that define clusters in high-dimensional data without access to prior knowledge. Typically, clustering

algorithms use a global objective function which implicitly assumes specific cluster structures in data [Duda *et al.*, 2001; Everitt *et al.*, 2001; Handl *et al.*, 2005; Theodoridis and Koutroumbas, 2009; Ultsch and Lötsch, 2017]. Moreover, cluster analysis has two additional challenges. For the clustering process, a wide variety of indices have been proposed to find the optimal number of clusters [Charrad *et al.*, 2012] and one of many statistical approaches has to be selected to test for the clustering tendency or so-called clusterability [Adolfsson *et al.*, 2019; Thrun, 2020]. After an extensive review of algorithms of behavior-based systems in unsupervised machine learning, two interesting concepts are addressed here, called self-organization and swarm intelligence. Moreover, two missing links are identified: emergence [Goldstein, 1999; Ultsch, 1999] and game theory [Nash, 1951].

The irreducible structures of high-dimensional data can emerge through self-organization in a phenomenon called emergence. Exploiting the Nash equilibrium concept from game theory [Nash, 1950] through the use of a swarm of intelligent agents, the data-driven approach presented in this work can outperform the optimization of a global objective function in the tasks of clustering. This is demonstrated using a collection of datasets offering a variety of real-world challenges, such as outliers or density vs. distance-defined clusters [Thrun and Ultsch, 2020].

2 Methods

The algorithms of DBS consists of three modules: projection with Pswarm, visualization via a topographic map of projected points and clustering.

2.1 Pswarm

The term planar projection method is often used for one type of dimensionality reduction methods. The output of a projection method is a scatter plot of projected points. Many projection methods are characterized by an objective function that is optimized using gradient descent or a corresponding learning algorithm [Thrun, 2018]. The quality of the projection and, consequently, the visualization will critically depend on the similarity concept chosen as the basis of the objective function [Thrun, 2018].

Focusing projection methods first adapt to global structures, and as time progresses, structure preservation shifts

from global optimization to the preservation of local neighborhoods. Projections of this type (e.g., NerV, CCA, ESOM, t-SNE) usually require parameters to be set because this phase, which is also called the learning phase, requires an annealing scheme. This task is challenging if no prior knowledge about the data exists.

In contrast to all other conventional projection methods, Pswarm neither does have any global objective function nor requires any input parameters other than the data set of interest. In this case, Euclidean distances are used in the input space. Alternatively, a user may also provide Pswarm with a matrix defined in terms of a particular dissimilarity measure, which is typically a distance but may also be a non-metric measure.

The intelligent agents of Pswarm, called DataBots [Ultsch, 2000] operate on a toroid grid, where positions are coded into polar coordinates to allow for the precise definition of their movement, neighborhood function, and annealing scheme. The size of the grid and, in contrast to other focusing projection methods, the annealing scheme are data-driven. During learning, each agent moves across the grid or stays in its current position in the search for the most potent scent emitted by other DataBots. Hence, agents search for other agents carrying data with the most similar features to themselves with a data-driven decreasing search radius. The movement of every agent is modeled using a game-theory approach, and the radius decreases only if a Nash equilibrium is found [Nash, 1950]. After the self-organization of agents is finished, the output of the Pswarm algorithm is a scatter plot of projected points.

2.2 Topographic Map

The goal of this scatter plot is a visualization of distance and density-based structures, which is often used in cluster analysis [Everitt *et al.*, 2001; Hennig, 2015; Mirkin, 2005; Ritter, 2014]. However, it is stated by the Johnson–Lindenstrauss lemma [Dasgupta and Gupta, 2003] that the two-dimensional similarities in the scatter plot cannot coercively represent high-dimensional structures. For example, similar data points can be mapped onto far-separated points, or a pair of closely neighboring positions represents a pair of distant data points.

Therefore, the generalized U-matrix [Thrun, 2018; Ultsch and Thrun, 2017] is exploited on this projection in the second step using emergence through an unsupervised artificial neural network called a simplified (because parameter-free) emergent self-organizing map. The generalized U-matrix generates the visualization of a topographic map with hypsometric tints, which can be vividly described as a virtual 3D landscape with a specific color scale chosen with an algorithm defining the contour lines [Thrun *et al.*, 2016]. The topographic map addresses the central problem in clustering, i.e., the correct estimation of the number of clusters. It allows the assessment of the number of clusters [Thrun *et al.*, 2016] by inspecting the 3D landscape. The color scale and contour lines imitate valleys, ridges, and basins: blue colors indicate small distances (sea level), green and brown colors indicate middle distances (low hills), and shades of gray and white indicate vast distances (high mountains covered with snow and ice). Valleys and basins represent clusters, and the watersheds of hills and mountains represent the borders between clusters. In this 3D landscape, the borders of the visualization are cyclically connected with a periodicity defined by two parameters (L,C).

2.3 Clustering

The semi-automated clustering is performed by calculating the shortest paths [Dijkstra, 1959] of the Delaunay graph between all projected points weighted with high-dimensional distances. This is possible because it was shown that the U-matrix is an approximation of the abstract U-matrix [Lötsch and Ultsch, 2014], which is based on Voronoi cells. Voronoi cells define a Delaunay graph where the edges between every projected point are weighted by the high-dimensional distances of the corresponding data points.

The clustering approach itself involves one of two choices. For each choice, a dendrogram can be visualized, which shows the ultrametric portion of the distance used is visualized (c.f. [Murtagh, 2004]). Large changes in fusion levels of the ultrametric portion of the distance indicate the best cut, but the resulting clustering should always be evaluated by the topographic map.

2.4 Open Source Access

There is a general need for open-source implementations in swarm intelligence algorithms [Martens *et al.*, 2011]. Thus, DBS is available as the R package "DatabionicSwarm" on CRAN (https://CRAN.R-project.org/package= DatabionicSwarm). Datasets are available in [Thrun and Ultsch, 2020]. The top 50 clustering algorithms are summarized in the R package "FCPS" on CRAN (https://CRAN. R-project.org/package=FCPS). A small subset of algorithms was selected for benchmarking in this work because for this subset the implicit assumptions were known in literature.

In the following section, the authors provide an exemplary usage of the algorithms based on data of 212 countries about the COVID-19 pandemia. The measured features of the COVID-19 virus and today's data is accessible in the Worldometer's COVID-19 (https://www.worldometers. info/coronavirus/). It should be noted that the data has a high amount of uncertainty because not all countries have the same reporting system or transparency. The source code, data extracted at 16.April 2020, and all analysis steps are accessible in https://zenodo.org/badge/latestdoi/257287298.

3 Exemplary Result of DBS

The following result serves as an illustration of the DBS algorithms. One topographic map is shown in Figure 1. Each point is labeled by a color that defines the cluster in which the point lies in. In this example, each point represents a country.

Figure 2 presents the evaluation of the clustering using a heatmap of distances ordered by the clustering. It is visible that similar countries are in a cluster and more dissimilar countries between clusters. Further, external quality assessment should be performed. Exemplary, it is shown that the clustering is coherent with regards to the spatial distribution of countries per clusters in Figure 3 which was generated using the R package 'DataVisualizations' (https://
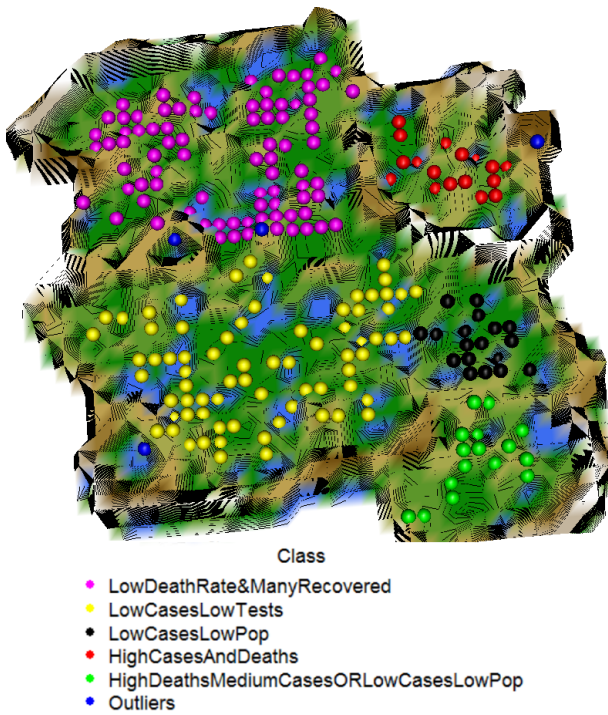
Figure 1: Topographic map of DBS projection and clustering using available data about the Covid-19 pandemic. Class names are derived by rules extracted from a decision tree.
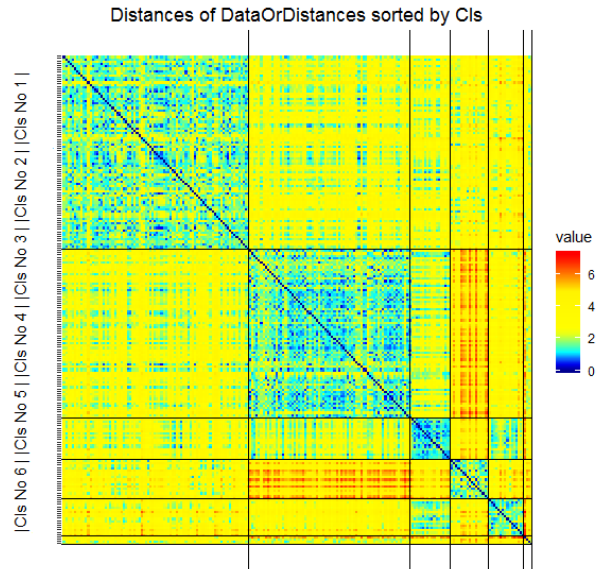

Figure 2: Heatmap of the distances and the clustering using in Figure 1. Blue colors indicate small distances and red and yellow colors large distances. Distances are ordered by the clustering.

CRAN.R-project.org/package=DataVisualizations) available on CRAN [Thrun and Ultsch, 2018]. The same colors color the world map in Figure 3 as the points in Figure 1. In this example, the clusters can be explained using decision trees leading to the class names provided in Figure 1 and Figure 3.

In the paper, many other real-word examples are presented with data of higher quality.

4 Exemplary Interpretation

The coexistence of high-dimensional structures visualization and clustering allows performing a valid cluster analysis even if the user is not an expert in clustering. DBS is able to find cluster structures for which countries "look like" each other but do not look much like objects outside the cluster (c.f. [Bonner, 1964]) even in this data with a high amount of uncertainty and noise. However, it remains a challenge to evaluate the cluster structures are meaningful. For example, the explanation of the magenta cluster (low death rate with high recovery rate) is questionable as it is more probable that these countries did not provide accurate data.

5 Pitfalls and Advantages in using DBS

Clustering algorithms possess the dilemma that they have to be simultaneously stable but exhibit plasticity allowing for the creation of new cluster structures [Duda *et al.*, 2001].

The benchmarking with conventional algorithms of Ward, single linkage, mixtures of Gaussians, k-means, spectral clustering, and PAM showed this dilemma. Clustering algorithms either have a small variance in results but are unable to reproduce many cluster structures or have a large variance and a smaller bias w.r.t cluster structures. For other machine learning approaches, this effect is well studied (e.g. [Geman *et al.*, 1992]). This work showed that clustering algorithms impose non-existent cluster structures on the data if their bias is large w.r.t. the data. In contrast, DBS enables the investigation of the cluster tendency and allows the user to improve the clustering if the topographic map is used interactively as shown in the attached source code.

In general, the bias of DBS is small, meaning that many different cluster challenges can be resolved, but the variance of results is sometimes considerable, meaning that in these cases, the result depends on the trial. As shown in the source code on the example above, in praxis, the user has the additional task to cut out an island from the toroidal topographic map (c.f. [Thrun *et al.*, 2016]).

The main pitfall of DBS is its computational complexity because for each row of data (high-dimensional data point), one additional agent (DataBot) has to be initialized, and the available open-source code lacks programming efficiency.

6 Conclusion

By exploiting the missing links between swarm-based algorithms and emergence as well as game theory, the main advantage of DBS is its robustness regarding very different types of distance and density-based structures of clusters. As a technique that uses swarm intelligence, DBS clustering is more robust with respect to outliers than conventional algorithms. DBS enables even a non-professional in the field of data mining to integrate its algorithms for visualization and/or clustering in their knowledge discovery process because no prior knowledge about the data is required, and no implicit
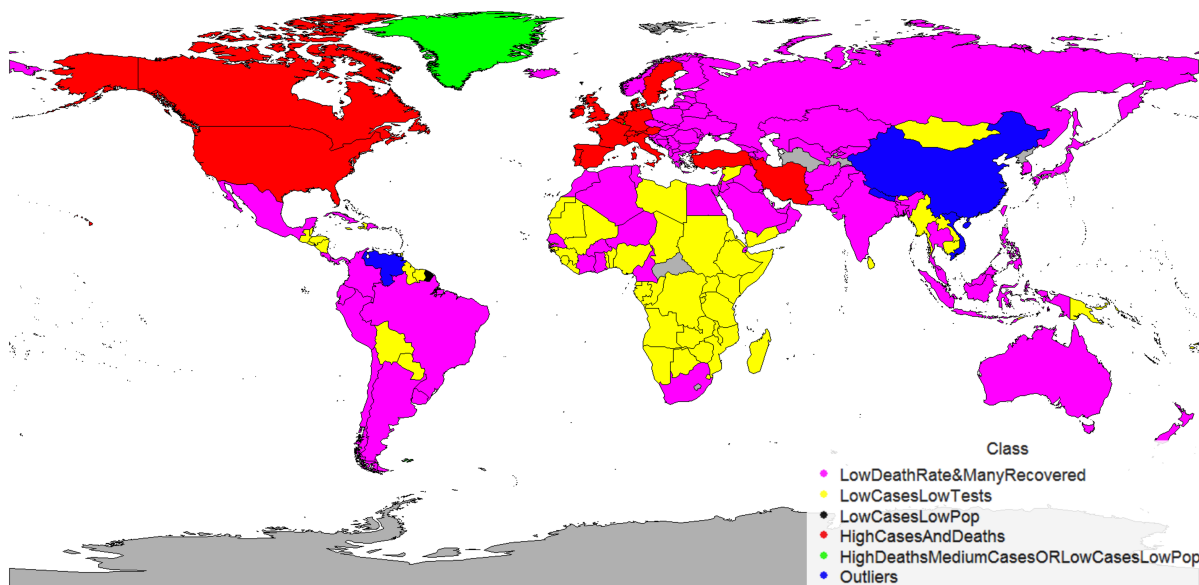
Figure 3: Map of the world with countries or regions colored by the DBS clustering of data about Covid-19 pandemic. Class names are based on a decision tree using the clustering of DBS. Colors are the same as in Figure 1. Black and green countries or regions are mostly islands. Blue countries are outliers.

assumptions about the data are made.

References

[Adolfsson *et al.*, 2019] Andreas Adolfsson, Margareta Ackerman, and Naomi C Brownstein. To cluster, or not to cluster: An analysis of clusterability methods. *Pattern Recognition*, 88:13–26, 2019.

[Bonner, 1964] R. E. Bonner. On some clustering technique. *IBM Journal of Research and Development*, 8(1):22–32, 1964.

[Charrad *et al.*, 2012] Malika Charrad, Nadia Ghazzali, Véronique Boiteau, and Azam Niknafs. Nbclust package: finding the relevant number of clusters in a dataset. *Journal of Statistical Software*, 61(6):1–36, 2012.

[Dasgupta and Gupta, 2003] Sanjoy Dasgupta and Anupam Gupta. An elementary proof of a theorem of johnson and lindenstrauss. *Random Structures & Algorithms*, 22(1):60–65, 2003.

[Dijkstra, 1959] Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, 1959.

[Duda *et al.*, 2001] Richard O Duda, Peter E Hart, and David G Stork. *Pattern Classification*. A Wiley-Interscience Publication. John Wiley & Sons, Ney York, USA, second edition, 2001.

[Everitt *et al.*, 2001] Brian S Everitt, Sabine Landau, and Morven Leese. *Cluster Analysis*. Arnold, London, fourth edition, 2001.

[Fayyad *et al.*, 1996] Usama M Fayyad, Gregory Piatetsky-Shapiro, Padhraic Smyth, and Ramasamy Uthurusamy. *Advances in knowledge discovery and data mining*, volume 21. American Association for Artificial Intelligence press, Menlo Park, California, USA, 1996.

[Geman *et al.*, 1992] Stuart Geman, Elie Bienenstock, and René Doursat. Neural networks and the bias/variance dilemma. *Neural Computation*, 4(1):1–58, 1992.

[Goldstein, 1999] Jeffrey Goldstein. Emergence as a construct: History and issues. *Emergence*, 1(1):49–72, 1999.

[Handl *et al.*, 2005] Julia Handl, Joshua Knowles, and Douglas B Kell. Computational cluster validation in postgenomic data analysis. *Bioinformatics*, 21(15):3201–3212, 2005.

[Hennig, 2015] et al. (eds.) Hennig, Christian. *Handbook of cluster analysis*. Handbook of Modern Statistical Methods. Chapman & Hall/CRC Press, New York, USA, 2015.

[Lötsch and Ultsch, 2014] Jörn Lötsch and Alfred Ultsch. Exploiting the structures of the u-matrix. In Th. Villmann, F.-M. Schleif, M. Kaden, and M. Lange, editors, *Advances in Self-Organizing Maps and Learning Vector Quantization*, pages 249–257. Springer International Publishing, June 2014.

[Martens *et al.*, 2011] David Martens, Bart Baesens, and Tom Fawcett. Editorial survey: swarm intelligence for data mining. *Machine Learning*, 82(1):1–42, 2011.

[Mirkin, 2005] Boris G. Mirkin. *Clustering: a data recovery approach*. Computer Science and Data Analysis Series. Chapnman & Hall/CRC, Boca Raton, FL, USA, 2005.

[Murtagh, 2004] Fionn Murtagh. On ultrametricity, data coding, and computation. *Journal of Classification*, 21(2):167–184, 2004.

[Nash, 1950] John Forbes Nash. Equilibrium points in n-person games. *Proc. Nat. Acad. Sci. USA*, 36(1):48–49, 1950.

[Nash, 1951] John Forbes Nash. Non-cooperative games. *Annals of Mathematics*, pages 286–295, 1951.

[Ritter, 2014] Gunter Ritter. *Robust cluster analysis and variable selection*. Monographs on Statistics and Applied Probability. Chapman & Hall/CRC Press, Passau, Germany, 2014.

[Theodoridis and Koutroumbas, 2009] Sergios Theodoridis and Konstantinos Koutroumbas. *Pattern Recognition*. Elsevier, Canada, fourth edition, 2009.

[Thrun and Ultsch, 2018] Michael C. Thrun and Alfred Ultsch. Effects of the payout system of income taxes to municipalities in germany. In Monika Papież and Sławomir Śmiech, editors, *12th Professor Aleksander Zelias International Conference on Modelling and Forecasting of Socio-Economic Phenomena*, pages 533–542, May 2018.

[Thrun and Ultsch, 2020] Michael C. Thrun and Alfred Ultsch. Clustering benchmark datasets exploiting the fundamental clustering problems. *Data in Brief*, 30(C):105501, 2020.

[Thrun *et al.*, 2016] Michael C. Thrun, F Lerch, Jörn Lötsch, and Alfred Ultsch. Visualization and 3d printing of multivariate data of biomarkers. In Vaclav Scala, editor, *International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG)*, volume 24, pages 7–16, May 2016.

[Thrun, 2018] Michael C. Thrun. *Projection Based Clustering through Self-Organization and Swarm Intelligence*. Springer, Heidelberg, 2018.

[Thrun, 2020] Michael C. Thrun. Improving the sensitivity of statistical testing for clusterability with mirrored-density plot. In Ian Archambault Daniel, Nabney and Peltonen Jaakkor, editors, *Machine Learning Methods in Visualisation for Big Data (MLVis)*, pages 1–5. The Eurographics Association, May 2020.

[Ultsch and Lötsch, 2017] Alfred Ultsch and Jörn Lötsch. Machine-learned cluster identification in high-dimensional data. *Journal of Biomedical Informatics*, 66(C):95–104, 2017.

[Ultsch and Thrun, 2017] Alfred Ultsch and Michael C. Thrun. Credible visualizations for planar projections. In Marie Cottrell, editor, *12th International Workshop on Self-Organizing Maps and Learning Vector Quantization, Clustering and Data Visualization (WSOM)*, pages 1–5, June 2017.

[Ultsch, 1999] Alfred Ultsch. *Data mining and knowledge discovery with emergent self-organizing feature maps for multivariate time series*, pages 33–46. Elsevier, first edition, 1999.

[Ultsch, 2000] Alfred Ultsch. Clustering with databots. In *nt. Conf. Advances in Intelligent Systems Theory and Applications (AISTA)*, pages 99–104, Feb 2000.