COMPARISON OF TIME-SERIES CLUSTERING SUCCESS

Karolína Bakuncová – Tomáš Löster

Abstract

The paper tries to outline and present practical usage of time-series clustering on different types of datasets by evaluating the overall success and quality of results. For the purposes of the paper, five datasets from the UCR archive have been selected. Clustering itself is done using the TSclust package in R. Based on findings it is possible to conclude, that the results of clustering differ significantly across all five datasets. The most successful hierarchical method of clustering was furthest neighbour (highest similarity coefficient 0,7705 and ARI 0,5592) while most successful distance measure is DTW (highest similarity coefficient 0,5400 and ARI 0,2956). For partitional methods the results were very comparable with hierarchical methods. In the case of finer differences between time series, the examined clustering methods were not able to reliably distinguish between them. Quality coefficients often provide questionable conclusions, leading to the necessity to verify their conclusions with a closer examination of the resulting distribution of clustered objects.

Key words: time-series, clustering

JEL Code: C38

Introduction

Cluster analysis has always been one of the most useful tools for discovering hidden information contained in data, including complex structures and relationships between variables that may go undetected to the naked eye. By identifying these underlying groups within given data, we are able to also derive their characteristics and other useful observations. On the other hand, this also allows for the detection of outliers and other information outside the known norm. All in all, this tool is favoured for both its versatility and relative simplicity in interpretation. The basic principle of cluster analysis can be defined as two key points:

- choosing an appropriate distance measure that is compatible with the given clustering algorithm,

- definition of groups and their subsequent quality assessment, either in terms of homogeneity or agreement of categorization based on known group assignment.

Additionally, there has been an increase in interest pertaining the investigation of characteristics of given data not only in terms of their current state, but also in terms of their variability over time. Due to this factor, advances in data analysis have created a demand for methods that are able to accommodate such needs. This concept becomes even more complicated if we consider time-series, which are notorious for their specific nature in terms of data analysis, e.g. volatility or autocorrelation (Mori, Mendiburu, & Lozano, 2016).

One of the possible solutions is a combination of procedures and ideas in the application of basic cluster analysis modified by the needs of tracking data over time. That is, creating another branch of cluster analysis, where the procedures, whether from the point of view of the calculation of similarity measures or creating group representatives, are applicable to time-series.

In practice, time-series clustering can be encountered stock analysis or prediction of sales for a newly introduced product. Generally, examples for clustering time-series can be found in many fields, including economics, finance, medicine, ecology, environmental studies, engineering, and many others. To this end, popular programming languages such as R or Python have adapted to this need and introduced packages designed for time-series clustering.

1 Data

The datasets used in this paper come from an archive submitted by the University of California, Riverside (Keogh, et al., 2006). It is a publicly available database of 129 datasets containing labelled time-series with the goal of improving time-series classification algorithms.

Five datasets were randomly selected to compare in terms of quality of classification using several classification methods implemented in R and the subsequent suitability of individual distances and methods for different types of time-series. Below are short descriptions for all relevant datasets:

- **ProximalPhalanxOutlineAgeGroup** – data contain information pertaining to the prediction of age of the monitored subjects based on the contours of their phalanx.

The goal of the classification task is then to classify the subject's image into its correct age category: 0 to 6 years, 7 to 12 years or 13 to 19 years (Davis, 2013).

- **Yoga** – dataset is based on images of the transition between yoga poses for two different subjects, male and female. The data was obtained by measuring the distance between the detected contour and the predefined centre.
- **MedicalImages** – dataset represents histograms of the pixel density for health images. The individual categories are based on parts of the body, a total of ten unique groups.
- **SwedishLeaf** – time-series represents the outlines of the leaves for Swedish trees. The individual categories in the assemblage are based on different tree types: Ulmus carpinifolia, Acer, Salix aurita, Quercus, Alnus incana, Betula pubescens, Salix alba 'Sericea', Populus tremola, Ulmus glabra, Sorbus aucuparia, Salix sinerea, Populus, Tilia, Sorbus intermedia and Fagus silvatica (Söderkvist, 2001).
- **Fish –** time-series represent fish contours categorized by species (Lee, et al., 2008).

2 Coefficients

For data clustering, quality can be usually determined based on two points of view, either by comparing the newly created clusters to their original label or evaluating the quality of clustering based on the similarity of individual objects within the created clusters. One of the tools used for this verification are quality coefficients, which are metrics that allow for the quantification of quality assessment. The coefficients can also be used to compare the success rate between several combinations of clustering procedures. For the purposes of this paper, the chosen quality coefficients are **Augmented Rand index** and **Similarity index** integrated into the R package TSClust.

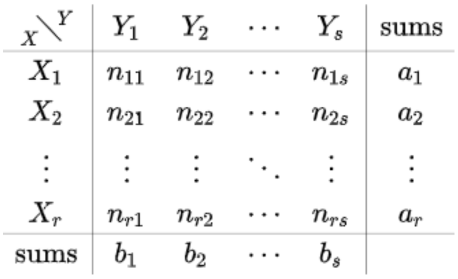
Adjusted Rand index (Hubert & Arabie, 1985)

The change in the calculation of ARI against its unmodified version is to consider that the agreement between two clustering methods may arise by chance. The calculation is thus as follows:

$$ARI = \frac{\Sigma_{ij}\binom{n_{ij}}{2} - \left[\Sigma_i\binom{a_i}{2}\Sigma_j\binom{b_j}{2}\right]/\binom{n}{2}}{\frac{1}{2}\left[\Sigma_i\binom{a_i}{2} + \Sigma_j\binom{b_j}{2}\right] - \left[\Sigma_i\binom{a_i}{2}\Sigma_j\binom{b_j}{2}\right]/\binom{n}{2}}, \quad (1)$$

where $i = 1, \ldots, r$ and $j = 1, \ldots, s$. Values n_{ij}, a_i, and b_j can be used from a pivot table created by comparing the resulting clusters for both methods (**Fig. 1**).

Fig. 1: Pivot table for the calculation of ARI



$X \diagdown Y$	Y_1	Y_2	$\cdots$	Y_s	sums
X_1	n_{11}	n_{12}	$\cdots$	n_{1s}	a_1
X_2	n_{21}	n_{22}	$\cdots$	n_{2s}	a_2
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
X_r	n_{r1}	n_{r2}	$\cdots$	n_{rs}	a_r
sums	b_1	b_2	$\cdots$	b_s	

The ARI coefficient quantifies how much better the model is at categorizing observations compared to random chance. Unlike RI, which only takes values from the interval <0,1>, ARI can also achieve negative values. These values can't be interpreted directly, but they can express a problem with the implementation of the model.

Similarity coefficient (Montero & Vilar, 2014)

The index implemented within the TSClust package represents a measure that compares the agreement between the actual distribution of observations into the original clusters $G = \{G_1, \dots , G_\mathrm{k}\}$ that we know, and the clusters obtained as a result of the cluster analysis $A = \{A_1, \dots , A_k\}$. The similarity coefficient is further defined by the formula:

$$Sim(G,A) = \frac{1}{k}\sum_{i=1}^{k} \max_{1\le j\le k} Sim(G_i, A_j)\,, \qquad (2)$$

where

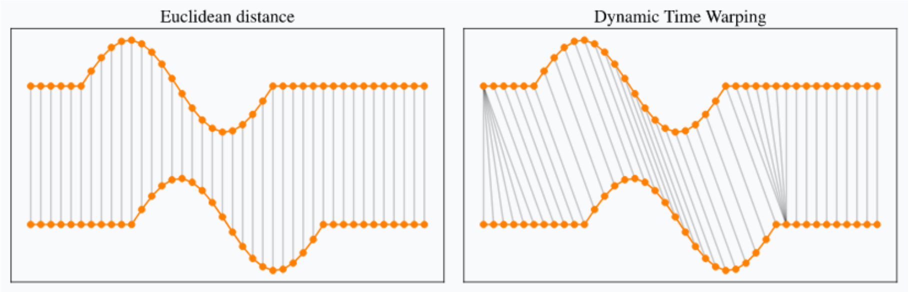
$$Sim(G_i, A_j) = \frac{|G_i \cap A_j|}{|G_i| + |A_j|}\,. \qquad (3)$$

3 Distance measures

There are several ways how to calculate the distance between clustering objects, either by taking the clustering objects directly or calculating the distance based on their transformations and other suitable characterizations.

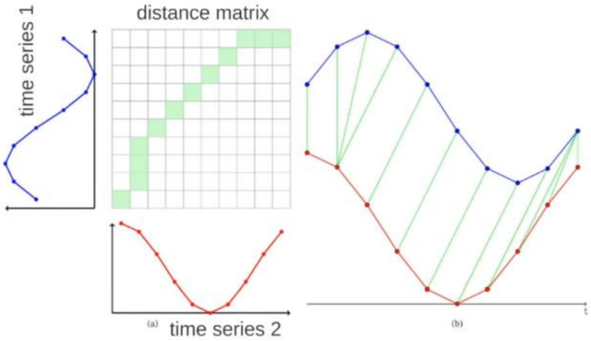
Dynamic time warping (DTW) is based on finding an optimal path between a pair of time-series by warping of the points assignment between the starting and final positions via temporal distortion (Müller, 2007) (**Fig. 2**). To achieve this, DTW uses a distance matrix between the time points of the given time-series (**Fig. 3**), where it strives to minimise the sum of distances in the matrix starting from the upper right corner to the bottom left corner of the matrix.

Fig. 2: Comparison of Euclidian and DTW distances



Source: https://rtavenar.github.io/blog/dtw.html

Fig. 3: Computations of DTW distance



Source: https://youtu.be/_K1OsqCicBY

Based on the calculation of DTW, it is possible to compute the distance between time-series of varying length. However, this method sometimes yields subpar results compared to the interpolation of time-series (Ratanamahatana & Keogh, 2004). For more information about DTW and its implementation in R see (Giorgino, 2009).

The principle of calculating **Shape Based Distance (SBD)** is based on cross-correlation, which is used to measure the degree of similarity between two time-series. An important condition for this distance measure is the need to normalize the time-series beforehand.

In general, the given metric indicates the number of time-series shifts relative to each other and the linear dependence of their values. Cross-correlation makes it possible to quantify this degree of similarity, even if the values are not sorted correctly. The resulting cross-correlation takes the form $CC_w\ (\boldsymbol{X}_T, \boldsymbol{Y}_T) = (c_1, \ldots, c_w)$.

By normalizing the cross-correlation sequence (dividing by the geometric mean of the autocorrelation functions for both time-series marked as $R_0\ (\boldsymbol{X}_T, \boldsymbol{X}_T)$ and $R_0\ (\boldsymbol{Y}_T, \boldsymbol{Y}_T)$) it is possible to derive the SBD distance formula:

$$d_{SBD}(\boldsymbol{X}_T,\boldsymbol{Y}_T) = 1 - \max_w\left(\frac{CC_w(\boldsymbol{X}_T,\boldsymbol{Y}_T)}{\sqrt{R_0(\boldsymbol{X}_T,\boldsymbol{X}_T)\,R_0(\boldsymbol{Y}_T,\boldsymbol{Y}_T)}}\right). \tag{4}$$

The authors of the paper *k-Shape: Efficient and Accurate Clustering of Time-series* (Paparrizos & Gravano, 2015) discuss SBD distance and the corresponding clustering algorithm in more detail.

Like the previous distances, the **ACF distance** also attempts to calculate the similarity between two clustered objects, this time considering the dependence of the time-series with each other. Instead of the time-series themselves, it uses their autocorrelation functions.

Assume that $\widehat{\boldsymbol{\rho}}_{X_T} = \left(\widehat{\rho}_{1,X_T},\dots,\widehat{\rho}_{L,X_T}\right)^T$ and $\widehat{\boldsymbol{\rho}}_{Y_T} = \left(\widehat{\rho}_{1,Y_T},\dots,\widehat{\rho}_{L,Y_T}\right)^T$ are autocorrelation functions for vectors $\boldsymbol{X}_T$ and $\boldsymbol{Y}_T$, where if $i > L$ then $\widehat{\rho}_{i,X_T} \approx 0$ and $\widehat{\rho}_{i,Y_T} \approx 0$. The ACF distance can be thus expressed as

$$d_{ACF}(\boldsymbol{X}_T,\boldsymbol{Y}_T) = \sqrt{\left(\widehat{\boldsymbol{\rho}}_{X_T} - \widehat{\boldsymbol{\rho}}_{Y_T}\right)^T \boldsymbol{\Omega}\left(\widehat{\boldsymbol{\rho}}_{X_T} - \widehat{\boldsymbol{\rho}}_{Y_T}\right)}, \tag{5}$$

where Ω is the matrix of weights (Galeano & Peña, 2000).

The definition of the matrix of weights Ω leads to two variations of the general formula:

- The matrix of weights is given as a unit matrix $\boldsymbol{\Omega} = \boldsymbol{I}$. Respectively, all values have the same weight so that the d_{ACF} represents the calculation of the Euclidean distance between the autocorrelation functions.

$$d_{ACFU}(\boldsymbol{X}_T,\boldsymbol{Y}_T) = \sqrt{\sum_{i=1}^{L}\left(\widehat{\boldsymbol{\rho}}_{i,X_T} - \widehat{\boldsymbol{\rho}}_{i,Y_T}\right)^2}. \tag{6}$$

- The weights of the values geometrically decrease towards the past, so that the calculation of d_{ACF} takes the following form:

$$d_{ACFG}(\boldsymbol{X}_T,\boldsymbol{Y}_T) = \sqrt{\sum_{i=1}^{L} p(1-p)^i\left(\widehat{\boldsymbol{\rho}}_{i,X_T} - \widehat{\boldsymbol{\rho}}_{i,Y_T}\right)^2}, \qquad 0 < p < 1, \tag{7}$$

where *p* represents the rate of geometric decrease of weights.

4 Clustering algorithm

It is possible to use classical clustering algorithms for time-series clustering. As such, the basic algorithms of time-series clustering can be divided into partitional and hierarchical, including further possible division into agglomerative and divisional.

The difference between the classical use of the given algorithms and their modification to time-series lies not in the use of the algorithms themselves, but in the modification of the time-series before their application. In other words, an important role lies with the selection of the appropriate distance measure suitable for dynamic data or their transformations (Liao, 2005).

Complete linkage

Unlike the nearest neighbour method, the farthest neighbour method, as the name implies, works with the highest possible distances between pairs of observations. More precisely, the algorithm takes the largest possible distance between a pair of observations, each of which being representatives of two different clusters. By finding the largest possible distances of a cluster relative to all other clusters, the smallest of these distances is combined. The clusters formed are usually spherical in their shapes and are more robust to the issue of chaining.

Average linkage method

The similarity between two clusters is determined by the average distance between all pairs of objects, each of which belongs to a different cluster. Merging of clusters is carried out on the basis of the smallest distance obtained. The use of the average distance results in a more robust method with respect to outliers. Adding a new observation does not have to significantly affect the result of clustering if the given probability distribution is respected. At the same time, the average bond method is less prone to ambiguous results (there is often no situation where the distance between several clusters is the same).

DTW barycenter averaging

The basis of the DBA method is a heuristic algorithm representing a global averaging function. In individual iterations, the representative of the group mean is optimized in a way that leads to minimizing the DTW distance of other group members to their group mean.

The minimization of the distance to the group mean can be expressed by minimizing the sum of square DTW distances of other sequences in the group to the total mean.

Due to the DTW distance, the optimization of individual points of the average group sequence becomes more complicated due to time distortion. The principle of DBA is based on the calculation of the points of an average sequence as a sequence of barycenters of coordinates associated with them. Minimizing the proportion of individual coordinates in the total weighted sum of distances leads to a decrease in the total sum of distances.

A detailed description of the DBA algorithm can be found in the paper *A global averaging method for dynamic time warping, with applications to clustering* (Petitjean, et al., 2011).

K-Shape clustering

The k-Shape algorithm is based on a similar idea to the k-means method, where optimal group means are formed over several iterations. In individual iterations, the sum of squares of distances is minimized, leading to the formation of internally homogeneous and mutually distinguishable clusters. Advantage of the k-Shape method is the possibility of linear scaling with the increase in the number of time-series.

In general, it is a method that can effectively compare sequences with each other and at the same time allows the calculation of centroids under invariance with respect to shifting, scaling and other time manipulations.

A detailed description of the k-Shape method can be found in the same publication as the SBD distance description used in the method (Paparrizos & Gravano, 2015).

5 Results

The best results for the combinations of partitional and hierarchical methods are compiled in **Tab. 1**. Based on the gathered information we can conclude that the overall quality of clustering is heavily based upon the not only the chosen combination of a method but the quality of data as well.

Similarity coefficient for all of the chosen datasets usually does not show a significant difference in clustering quality between hierarchical and partitional methods, except datasets ProximalPhalanxOutlineAgeGroup and Fish. Curiously, in certain cases we can see a noticeable disagreement about the final quality of clusters when using similarity coefficient and ARI. For example, in the dataset Yoga, similarity coefficient is relatively high (more than 0,50), even though the ARI coefficient is contrarily quite low, especially for the chosen partitional method (0,0031). On the other hand, for dataset SwedishLeaf the similarity coefficient is slightly lower (more than 0,40) but ARI is much higher (more than 0,20). We can thus come to the realisation that we cannot rely on coefficient alone when evaluating the quality of clustering. One of the other possible tools for evaluating the quality of clusters are visualisations, either via graphical tools or tables.

Tab. 1: Results of the clustering analysis

Dataset	n	k	Algorithm	Distance	S	ARI
ProximalPhalanxOutlineAgeGroup	605	3	AVERAGE	ACF	0,7705	0,5592
			SHAPE	SBD	0,6785	0,5256
Yoga	3 300	2	COMPLETE	ACF	0,5758	0,0266
			DBA	DTW	0,5400	0,0031
MedicalImages	1 141	10	COMPLETE	DTW	0,3666	0,0693
			DBA	DTW	0,3283	0,0690
SwedishLeaf	1 125	15	COMPLETE	ACF	0,4024	0,2169
			SHAPE	SBD	0,4333	0,2873
Fish	463	7	COMPLETE	DTW	0,3920	0,2035
			DBA	DTW	0,5110	0,2956

n ... number of time-series, k ... number of groups, S ... similarity coefficient

Looking at the results of hierarchical clustering, based on figures **Fig. 4** to **Fig. 8** it is evident, that while some datasets allow for the creation of visually separated and homogenous groups, other dataset show a noticeable difficulty in that regard. However, it is important to note that in some of these cases, for example dataset Yoga, the initial assignment of time-series to their respective labels already was not homogenous, which further complicates the creation of a quality model. Moreover, some of the groups are visibly similar to each other, which often times leads to an incorrect assignment of members, especially in the case of distance measures focusing on the comparison of shapes (ex. **Fig. 4**).

Fig. 4: Hierarchical clustering of ProximalPhalanxOutlineAgeGroup

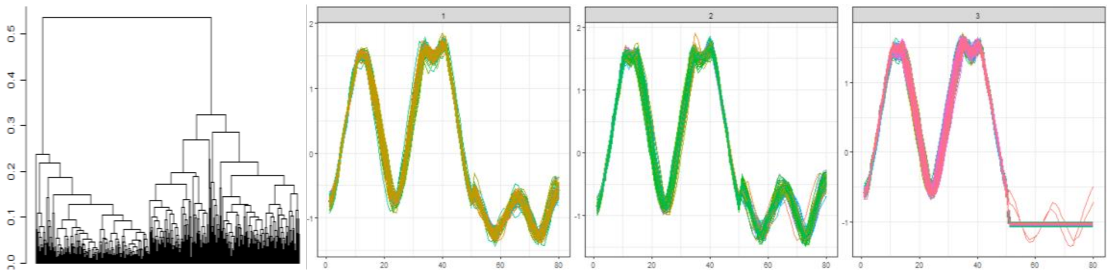


Fig. 5: Hierarchical clustering of Yoga

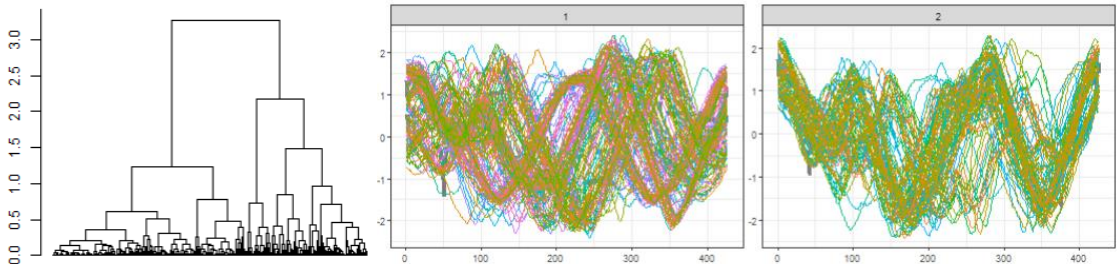


Fig. 6: Hierarchical clustering of MedicalImages

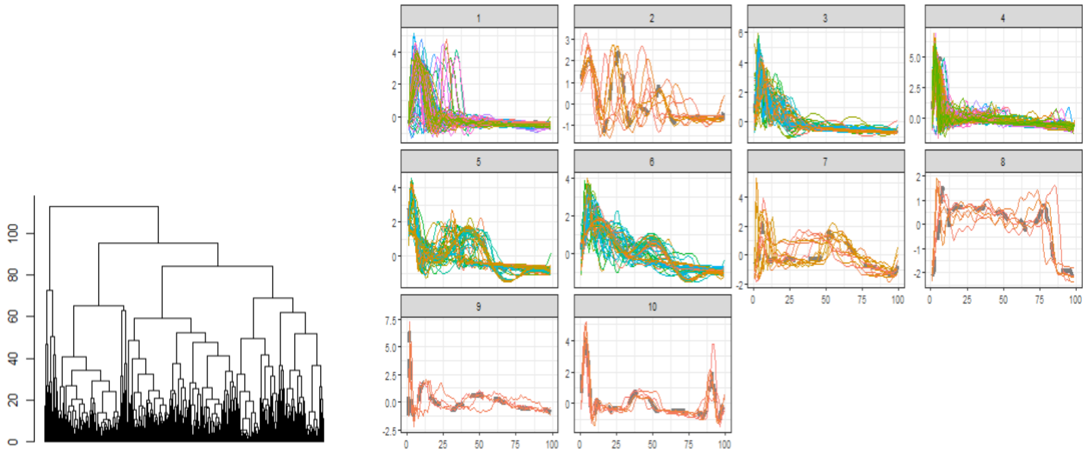


Fig. 7: Hierarchical clustering of SwedishLeaf

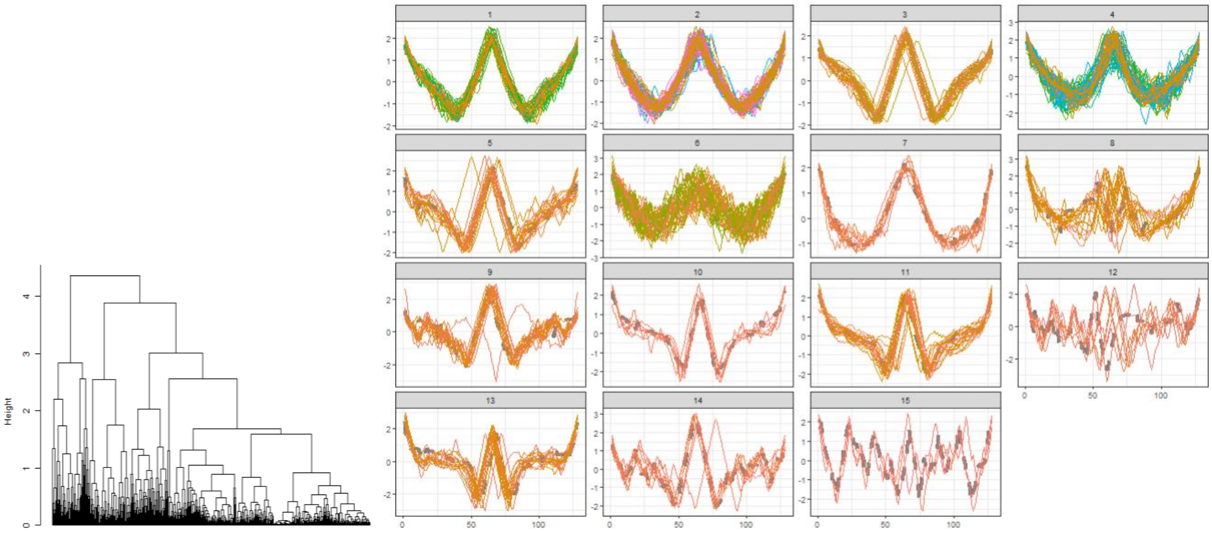
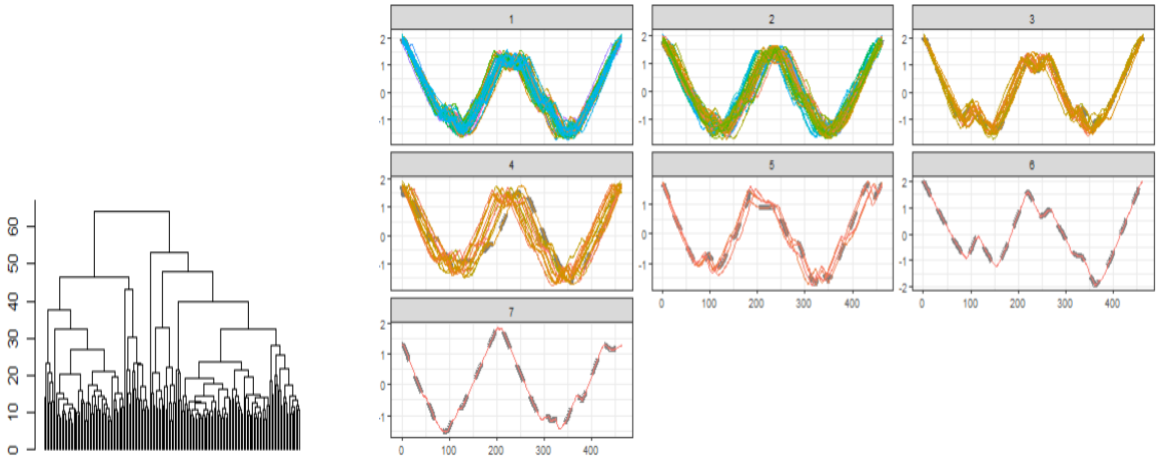


Fig. 8: Hierarchical clustering of Fish



The results of partitional clustering are in most cases similar to the results of hierarchical clustering (**Fig. 9 to Fig. 13**). However, there are two cases where we can see some significant differences. First, chosen partitional clustering methds performed noticeably worse in the case of dataset ProximalPhalanxOutlineAgeGroup (similarity coefficient for hierarchial methods reached 0,77 while for partitional only 0,68). On the other hand, partitional methods performed significantly better in the case of datsate Fish, where similarity coefficient reached the value of 0,51 (compared to 0,39 for hierarchical methods).

Fig. 9: Partitional clustering of ProximalPhalanxOutlineAgeGroup

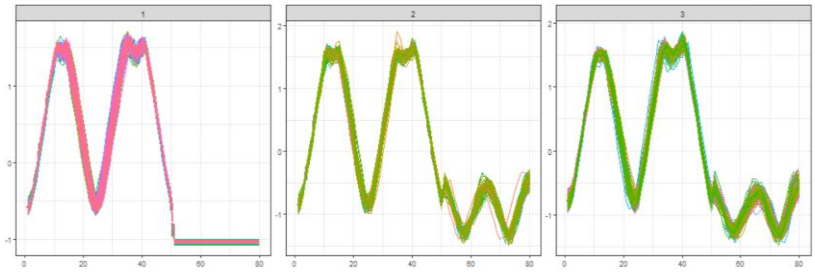


Fig. 10: Partitional clustering of Yoga

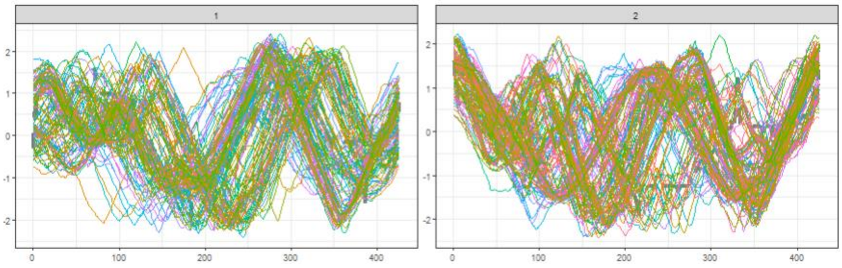


Fig. 11: Partitional clustering of MedicalImages

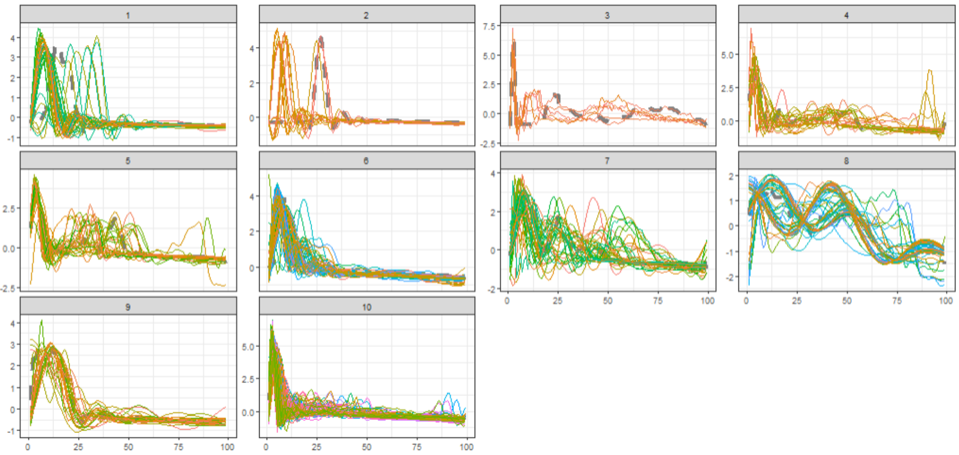


Fig. 12: Partitional clustering of SwedishLeaf

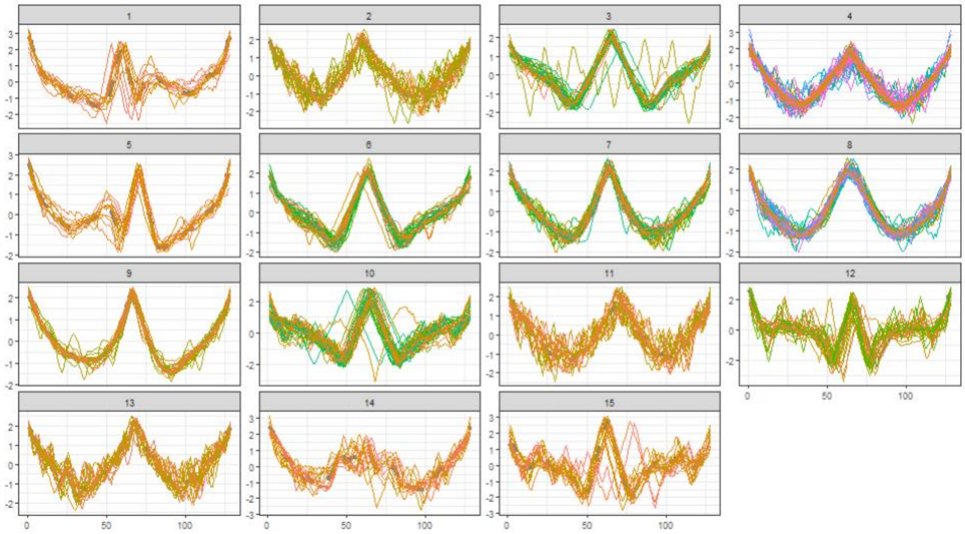
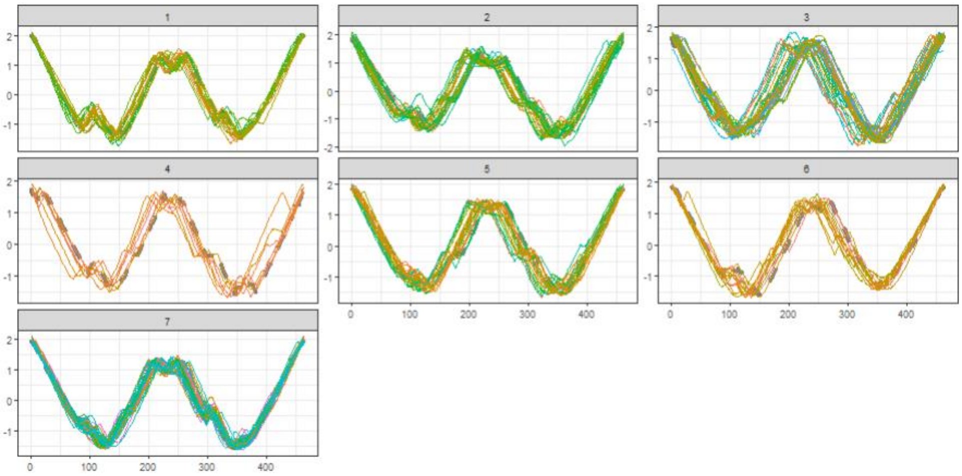


Fig. 13: Partitional clustering of Fish



Conclusion

The objectively best combination of methods was chosen based on preliminary information gathered from the coefficients, which was then verified by using several visualisation methods, to further gain more insight into the clustering, such as dendrograms, graphs of final clusters, comparison of cluster internal structures and their actual assignment, etc.

The quality of clustering varied significantly based on both the quality of the used dataset and the chosen combination of methods. In most cases, the results gained were mostly agreeable, although there is much room for improvement. It is important to not rely on just quality coefficients for evaluation, as time-series are much more nuanced, and the coefficients are not able to always accurately capture the reality of the situation.

References

Davis, L. (2013). Predictive modelling of bone ageing. Norwich: University of East Anglia.

Galeano, P., & Peña, D. (2000). Multivariate Analysis in Vector Time Series. *Resenhas do Instituto de Matemática e Estatística da Universidade de São Paulo, 4*(4), 383–403.

Giorgino, T. (2009, August). Computing and Visualizing Dynamic Time Warping. *Journal of Statistical Software, 31*(7), 1–24.

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification, 2*(1), 193–218.

Keogh, E., Wei, L., Xi, X., Lonardi, S., Sheih, J., & Sirowy, S. (2006). Intelligent Icons: Integrating Lite-Weight Data Mining and Visualization into GUI Operating Systems. *Sixth International Conference on Data Mining (ICDM'06)* (pp. 912-916). Hong Kong, China: IEEE. doi:10.1109/ICDM.2006.90

Lee, D., Archibald, J. K., Schoenberger, R. B., Dennis, A. W., & Shiozawa, D. K. (2008). Contour Matching for Fish Species Recognition and Migration Monitoring. In *Applications of Computational Intelligence in Biology* (pp. 183–207). Heidelberg, Germany: Springer Berlin.

Liao, T. W. (2005, November). Clustering of time series data—a survey. *Pattern Recognition, 38*(11), pp. 1857–1874.

Montero, P., & Vilar, J. (2014). TSclust: An R Package for Time Series Clustering. *Journal of Statistical Software, 62*(1), 1–43. Retrieved from http://www.jstatsoft.org/v62/i01/

Mori, U., Mendiburu, A., & Lozano, J. A. (2016, December). Distance Measures for Time Series in R: The TSdist Package. *The R Journal, 8*(2).

Müller, M. (2007). Dynamic Time Warping. In M. Müller, *Information Retrieval for Music and Motion* (pp. 69–84). Heidelberg, Germany: Springer Berlin.

Paparrizos, J., & Gravano, L. (2015). k-Shape: Efficient and Accurate Clustering of Time Series. *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data* (pp. 1855–1870). Masa, Thailand: SIGMOD '15.

Petitjean, F., Ketterlin, A., & Gançarski, P. (2011). A global averaging method for dynamic time warping, with applications to clustering. *Pattern recognition, 44*(3), pp. 678–693.

Ratanamahatana, C. A., & Keogh, E. (2004). Everything you know about dynamic time warping is wrong. *Third Workshop on Mining Temporal and Sequential Data* (pp. 1–11). Seattle, WA, USA: Citeseer.

Söderkvist, O. J. (2001). Computer vision classifcation of leaves from swedish trees. Linkoping, Sweden: Linkoping University.

Contact

Bakuncová Karolína, Ing.
Prague University of Economics and Business
W. Churchill Sq. 1938/4
130 67 Prague 3, Czech Republic
bakk02@vse.cz

Löster Tomáš, Ing., Ph.D
Prague University of Economics and Business
W. Churchill Sq. 1938/4
130 67 Prague 3, Czech Republic
losterto@vse.cz