Archimer
https://archimer.ifremer.fr

***In* Database and Expert Systems Applications. 32nd International Conference, DEXA 2021, Virtual Event, September 27–30, 2021, Proceedings, Part II. Lecture Notes in Computer Science, vol 12924. pp 232-238**
2021,
Eds Strauss C., Kotsis G., Tjoa A.M., Khalil I.
ISBN 9783030864750
https://doi.org/10.1007/978-3-030-86475-0_23
https://archimer.ifremer.fr/doc/00724/83583/

TSX-Means: An Optimal K Search Approach for Time Series Clustering

Tokotoko Jannai [1], Selmaoui-Folcher Nazha [1], Govan Rodrigue [1], Lemonnier Hugues [2]

[1] ISEA, University of New Caledonia, Nouméa, New Caledonia
[2] UMR ENTROPIE, IFREMER, Nouméa, New Caledonia

Email addresses : jannai.tokotoko@unc.nc ; nazha.selmaoui@unc.nc ; hugues.lemonnier@ifremer.fr

Abstract :

Proliferation of temporal data in many domains has generated considerable interest in the analysis and use of time series. In that context, clustering is one of the most popular data mining methods. Whilst time series clustering algorithms generally succeed in capturing differences in shapes, they most often fail to perform clustering based on both shape and amplitude dissimilarities. In this paper, we propose a new time series clustering method that automatically determines an optimal number of clusters. Cluster refinement is based on a new dispersion criterion applied to distances between time series and their representative within a cluster. That dispersion measure allows for considering both shape and amplitude of time series. We test our method on datasets and compare results with those from K-means time series (TSK-means) and K-shape methods.

Please note that this is an author-produced PDF of an article accepted for publication following peer review. The definitive publisher-authenticated version is available on the publisher Web site.

1 Introduction

Time series analysis is applied in many areas of business engineering, finance, economics, health care, etc. It serves various purposes such as subsequence matching, anomaly detection, pattern discovery, clustering, classification, etc. Our study focuses on time series clustering There are two main approaches for time series clustering. The first approach is based on feature construction. Series are described by a vector of feature attributes [5], and instances are grouped using a classical clustering method (*K-means*,*DBscan*, ...). The second one uses similarity measures adapted to time series comparison, combined with basic approaches (e.g. *K-means*) to cluster set of raw time series. Several similarity measures have been suggested for time series clustering, such that DTW [8], *SBD* [7], *LCSS* [10], and *ERP* [1] measures. All those distance measures compare series considering only effects of the temporal phase shift, and do not include amplitude drifts. However, in some application domains, time series clustering should be done by considering invariance and interval of measurements on the y axis as well as the shift of series on the x axis. Indeed, the range of values

* This work is supported by PIL (Province of Loyalty Islands) in New Caledonia

on the y axis can strongly discriminate between classes. For example, in agriculture or aquaculture domains, the range of y values in time series related to environmental data, such as changes in temperature, can significantly influences the growth and survival of living species. In this paper, we propose an approach based on shape analysis and that also takes into account the variance along the y axis. In addition, we develop a strategy that allows to automatically define an optimal number of clusters k using a new dispersion criterion applied on distances between instances and their representative within each cluster. Unlike most methods that normalize data, our approach can be applied to both normalized and raw time series. This new method is robust to the shifting of series on the x axis because we use metrics that take into account the distortion of series over time, in particular DTW, which is the most used for time series clustering [11, 3]. For the y shift, we consider a maximum interval over which those metrics vary. Section 2 presents notations and basic definitions. In section 3, we present our contribution, in which a new dispersion measure of distance distribution is presented as well as the principle of our method. Section 4 gives results of experiments on several datasets and compared to those of *TSK-means* [4] and *K-shape* [9].

2 Notations and definitions

Let s be a time series of length n where $s(i)$ corresponds to the value of the signal at time i. Let $T = \{s_1, s_2, \ldots, s_n\}$ a set of time series.

Clusters and their representatives: We call k-clustering C of T, the set $C = \{C_1, C_2, \ldots, C_k\}$ containing k homogeneous subsets of T (in relation to a measure of distance $Dist$), each having a representative noted R_{C_i} with $\forall i \in \{1, \ldots, k\}$, $C_i = \{s_{i_1}, s_{i_2}, \ldots, s_{i_{m_i}}\}$ and verifying the following criteria: **(1)** $T = \cup_{i=1}^{k} C_i$ and $C_i \cap C_j = \emptyset$ $\forall i \neq j$ and **(2)** $Dist(R_{c_i}, s) < Dist(R_{c_j}, s)$ $\forall s \in C_i$ and $j \neq i$. The representative of a cluster (called prototype) can be a centroid, medoid, etc.

Standard deviation and entropy of a cluster: Let C_i be a cluster of C on T according to a measure of distance $Dist$. Let $Dist(C_i) = \{d_{i_1}, \ldots, d_{i_{m_i}}\}$ the set of values of the $Dist$ between an instance of C_i and its representative R_{C_i}. Let $\sigma(C_i)$ the standard deviation calculated on the distribution of values taken by $Dist(C_i)$, and $E(C_i)$ its entropy measure. $\sigma(C_i) = \sqrt{\frac{1}{m_i}\sum_{k=1}^{m_i}(d_{i_k} - \overline{d_i})^2}$ where $\overline{d_i}$ is the average of $Dist(C_i)$ and $E(C_i) = -\sum_{k=1}^{m_i} P(d_{i_k}) \times \log(P(d_{i_k}))$. In this paper, we used the distance measure DTW optimizd by Kehog [6].

3 TSX-Means: A new method for time series clustering

Our approach mainly focuses on a new strategy for robust cluster refinement and automatic determination of the optimal number of clusters k. Any distance (or similarity) measure adapted to time series can be used in this approach. We tested it with different distance measures, such as measures derived from

DTW. The method, based on a minimum number of clusters initially set to *nb_min_clust* and a set of defined criteria, implements the principle of refining each cluster by revisiting all its instances. Instances that do not verify the criteria, in relation to the class they belong to, are put in a reject class. We then iterate the principle on that reject class (considered as a new set of series to be clustered) until the stopping conditions are verified. The criteria used in our approach are linked to the following thresholds: **(1)** *nb_min_inst*: the minimum number of instances allowed per cluster and **(2)** *seuil_disp*: the intra-cluster variability, defined from a new dispersion measure that depends on both the variability and the entropy measures of distances between each instance and its representative in cluster belongs to. In this contribution, we propose a new dispersion measure of distances between instances and their representative in a cluster. This dispersion measure, noted *disp*, is determined by the ratio between the standard deviation and the entropy of the distance values.

Definition 1 (measure of dispersion *disp*). *Let C_i a cluster of the set T. We define its measure of dispersion by: $disp(C_i) = \frac{\sigma(C_i)}{E(C_i)}$.*

If the dispersion is minimal then the homogeneity is maximal. $disp(C_i)$ reflects the inner cluster variability. The smaller *disp* is, the smaller the variability around the representative is. That allows to select the nearest instances to a representative according to a fixed threshold, denoted s_d in the following.

Criteria for selecting cluster instances: Let s_d a fixed threshold and C_i a cluster. A new associated cluster $C'_i \subset C_i$ is built, verifying the $disp(C'_i) \leq s_d$. Computation of the dispersion measure requires at least two values. A minimum number of instances initially in the new C'_i cluster is thus provided by *nb_min_inst* in the algorithm. In order to determine those instances, $Dist(C_i)$ are ordered and saved in $Sort(Dist(C_i)) = \{v_1, v_2, \ldots, v_m\}$ with $\forall\ i < j,\ v_i \leq v_j$ (procedure *ApplyCriteria*). We integrate in C'_i the first *nb_min_inst* instances in the sorted list $Sort(Dist(C_i))$. If $disp(C'_i) \leq s_d$ then other instances are added one by one in C'_i, as long as the criterion remains true, otherwise instances that do not verify the criterion are put in the reject cluster. The value $disp(C'_i)$ is updated each time an instance is added.

3.1 Principle of the method

The algorithm takes as parameters thresholds *nb_min_clust*, *nb_min_inst*, and s_d and uses any *Dist*. As output, it provides a number of clusters determined automatically based on dispersion criteria, and a reject class noted CR. The principle of the algorithm is the following:

Step 1: definition of initial clusters. Instances of T (set of time series) are partitioned into a minimum number of *nb_min_clust* clusters. To create those clusters, we apply the classic algorithm *TSK-Means* (or *K-shape*) with k = *nb_min_clust* and a distance measure *Dist* (f.ex *DTW*, etc.). The procedure $[C, Dist(C)] = CreateInitialsClusters(T, nb\_min\_clust)$ of algorithm 1 returns initial clusters.

Step 2: refining clusters by applying the dispersion criterion. The procedure $[C', CR] = ApplyCriteria(C, Dist, s_d, nb\_min\_inst)$ consists in applying the homogeneity criterion to each cluster C_i to only keep instances verifying that criterion. The remaining instances are assigned to the reject class CR. If the number of instances of an initial cluster C_i is less than nb_min_inst, then this cluster is deleted and its instances are assigned to the reject class.
Step 3: applying the stopping criterion. If the number of instances in the reject class is greater than nb_min_inst, then the initial step is repeated taking as new set T the rejected class. Otherwise, the algorithm stops.

3.2 *TSX-Means* algorithm

At first call of our recursive method (Algorithm 1), the number of clusters to be determined $nbClust$, is initialized to 0, and the set of final clusters C_f to the empty set. At each call of the recursive algorithm, a new set of at most nb_min_clust clusters and the reject cluster are created from initial clusters obtained by the $CreateInitialsClusters$ method. The algorithm is therefore repeated as long as the reject cluster is not empty and the number of instances is greater than nb_min_inst. The method could assign to the reject cluster CR the same instances indefinitely if no admitted new cluster C_f was generated. The $recursiveCpt$ iteration counter allows to stop the algorithm when it reaches a maximum number of iterations provided by the user. Thus, it is possible to get a number of clusters lower than nb_min_clust or even no cluster at all. This occurs when the *ApplyCriteria* method does not find any instance verifying the dispersion criterion in each of the initial clusters. This case is linked to a low value of the dispersion threshold. Nevertheless, increasing the threshold will integrate instances that are far from the representative and will lead to creating a cluster with high variability.

4 Experimental results

The method has been tested on data of the UEA & UCR [2] archives. We tested our algorithm on 20 datasets. Chosen datasets have series of various lengths and a different number of classes. Most of them have a low number of classes (≤ 7), we say non complex data. In order to test our new method TSX-Means on more complex data, the last 7 datasets have a higher number of classes (≥ 24). For each dataset and for each distance used, we tested our method by varying the parameters s_d and nb_min_clust. Nb_min_inst was set to the number of instances of the smallest class of the dataset. Once the number k is found by our algorithm, we run *TSK-means* and *k-shape* with the same value of k to compare the performances between the 3 methods. We used different metrics (Accuracy, ARI and V-Measure (VM)) for performance comparison averged for the tested parameters. Accuracy is calculated when the number of clusters is the actual number of classes. Otherwise, ARI and V-M are used. The parameter nb_min_clust has a greater impact on performance measures, and particularly

Algorithm 1 TSX-Means(T,nb_min_clust,s_d,$nbMaxIter$,$nbClust$,$recursifCpt$, nb_min_inst)

Output: - C_f set of clusters

if $nb\_min\_clust < p$ **then**
  $TmpCpt = 0$
  $\{C, Dist(C)\} = CreateInitialsClusters(T, nb\_min\_inst)$
  for i=1 to nb_min_clust **do**
    $\{C'$, $CR\}$ = ApplyCriteria(C_i, $Dist$, $seuil\_disp$, nb_min_inst)
    if $C' \neq \emptyset$ **then**
      $C_f[nbClust] = C'$
      $T = T - C'$
      $nbClust = nbClust + 1$
    else
      $TmpCpt = TmpCpt + 1$
    end if
  end for
  if $TmpCpt == nb\_min\_clust$ **then**
    $recursifCPT = recursifCpt + 1$
  end if
  if $recursifCpt < nbMaxIter$ **then**
    $TSX\text{-}Means(T, nb\_min\_clust, s_d, nbClust, nbMaxIter, recursifCpt)$
  end if
end if
return C_f

V-Measure, than threshold s_d. The difference of ARI and V-M are, in average, 10% higher for complex data for *TSX-Means* than for *K-Shape* method. The new dispersion measure is a good indicator of cluster homogeneity. In general, *TSX-Means* method is more efficient than other methods, especially when the number of classes is very high. Table 1 shows results for accuracy scores. We noticed that dispersion measure improves clustering performance. Indeed, *TSX-Means* outperforms *TSK-Means* and *K-Shape* methods for the majority of data. We noticed that the dispersion measure improves clustering performance with

Table 1. Accuracy of *TSX-Means* with initial clusters from *TSK-Means*.

dataset	Distances	k	TSX-Means	TSK-Means	Kshape	rejet TSX-means	Kmin
Car	sakoechiba	4	**0.446**	0.433	0.433	4	4
Fish	fast	7	**0.457**	0.440	0.391	0	5
Herring	itakura	2	**0.609**	0.594	0.509	0	2
LargeKitchen	sakoechiba	3	0.517	0.453	**0.521**	0	3
Meat	classic	3	**0.782**	0.653	0.750	1	3
Refrigeration	fast	3	**0.363**	0.361	0.360	0	3
SmallKitchen	itakura	3	0.417	**0.460**	0.407	0	3
WormsTwoClass	fast	2	**0.575**	0.511	0.602	0	2

the set of measures derived from DTW. Indeed, *TSX-Means* outperforms *TSK-Means* and *k-shape* methods for tha majority of data. Table 2 shows accuracy scores of *TSX-Means* using *K-Shape* as initial clusters generator. Our method outperforms *k-Shape* for 5/7 data.

Table 2. Accuracy of *TSX-Means* and the *K-Shape* with initial clusters from *K-Shape*

dataset	k	X-Shape	K-Shape	nb_min_clust
Computers	2	**0.616**	0.548	2
Meat	3	**0.700**	0.683	3
OSULeaf	6	0.420	0.435	6
OliveOil	4	**0.800**	0.433	4
RefrigerationDevices	3	**0.416**	0.368	3
ScreenType	3	0.357	0.360	3
Yoga	2	**0.487**	0.480	2

5 Conclusion and perspectives

We proposed a new dispersion measure in a cluster, and designed a new method *TSX-Means* for time series clustering, allowing to automatically determine an optimal number of clusters. This measure allows to refine clusters initially generated by existing clustering methods. Performance of *TSX-Means* was compared to *TSK-Means* and *K-Shape* methods on a set data. Quality measures of clustering performance showed that *TSX-Means* method outperforms *TSK-Means* and *K-Shape*, especially for data with a very large number of clusters.

References

1. Chen, L., Ng, R.T.: On the marriage of lp-norms and edit distance. In: VLDB. pp. 792–803. Morgan Kaufmann (2004)
2. Dau, H.A., Bagnall, A., Kamgar, K., Yeh, C.C.M., Zhu, Y., Gharghabi, S., Ratanamahatana, C.A., Keogh, E.: The ucr time series archive. IEEE/CAA Journal of Automatica Sinica **6**(6), 1293–1305 (2019)
3. Dilmi, D., Barthès, L., Mallet, C., Chazottes, A.: Iterative multiscale dynamic time warping (IMs-DTW): a tool for rainfall time series comparison. International Journal of Data Science and Analytics **10**, 65–79 (2020)
4. Huang, X., Ye, Y., Xiong, L., Lau, R.Y., Jiang, N., Wang, S.: Time series k-means: A new k-means type smooth subspace clustering for time series data. Information Sciences **367-368**, 1 – 13 (2016)
5. Kalpakis, K., Gada, D., Puttagunta, V.: Distance measures for effective clustering of arima time-series. In: ICDM. pp. 273–280 (2001)
6. Keogh, E., Pazzani, M.: Derivative dynamic time warping. First SIAM-ICDM **1**, 1–11 (2002)
7. Meesrikamolkul, W., Niennattrakul, V., Ratanamahatana, C.A.: Shape-based clustering for time series data. In: PaKDD. pp. 530–541 (2012)
8. Müller, M.: Dynamic time warping. Information retrieval for music and motion pp. 69–84 (2007)
9. Paparrizos, J., Gravano, L.: k-shape: Efficient and accurate clustering of time series. In: ACM SIGMOD ICMD. pp. 1855–1870 (2015)
10. Vlachos, M., Kollios, G., Gunopulos, D.: Discovering similar multidimensional trajectories. In: ICDE'02. pp. 673–684 (2002)
11. Zhang, Z., Tavenard, R., Bailly, A., Tang, X., Tang, P., Corpetti, T.: Dynamic time warping under limited warping path length. Information Sciences **393**, 91–107 (2017)