Behave or be detected!
Identifying outlier sequences by their group cohesion

Martha Tatusch, Gerhard Klassen, and Stefan Conrad

Heinrich Heine University, Universitätsstr. 1, 40225 Düsseldorf, Germany
{tatusch,klassen,stefan.conrad}@hhu.de

Abstract. Since the amount of sequentially recorded data is constantly increasing, the analysis of time series (TS), and especially the identification of anomalous points and subsequences, is nowadays an important field of research. Many approaches consider only a single TS, but in some cases multiple sequences need to be investigated. In 2019 we presented a new method to detect behavior-based outliers in TS which analyses relations of sequences to their peers. Therefore we clustered data points of TS per timestamp and calculated distances between the resulting clusters of different points in time. We realized this by evaluating the number of peers a TS is moving with. We defined a stability measure for time series and subsequences, which is used to detect the outliers. Originally we considered cluster splits but did not take merges into account. In this work we present two major modifications to our previous work, namely the introduction of the jaccard index as a distance measure for clusters and a weighting function, which enables behavior-based outlier detection in larger TS. We evaluate our modifications separately and in conjunction on two real and one artificial data set. The adjustments lead to well reasoned and sound results, which are robust regarding larger TS.

Keywords: Outlier Detection · Time Series Analysis · Clustering.

1 Introduction

With increasing understanding about the value of data and the rising amount of connected sensors in the world of the IoT, more data is recorded every day than ever before. This enables a time aware analysis of the accumulated data by regarding it as time series. The time-driven data view not only allows the extraction of trends and seasons but also an interpretation of behavior. This is especially the case when several time series are considered at the same time. In our paper [20] we introduced an outlier detection algorithm based on the relative behavior of time series. As this was a novel approach we were aware of some drawbacks and application specific requirements. In concrete we noticed that earlier clusters had a high impact and that a cluster split would not be treated the same way as a cluster merge. While the latter is an application dependent circumstance the first causes a high dependence on early points in time, which is not wanted in most cases. In order to overcome these drawbacks we now introduce a weighting function and a new way of calculating the cluster

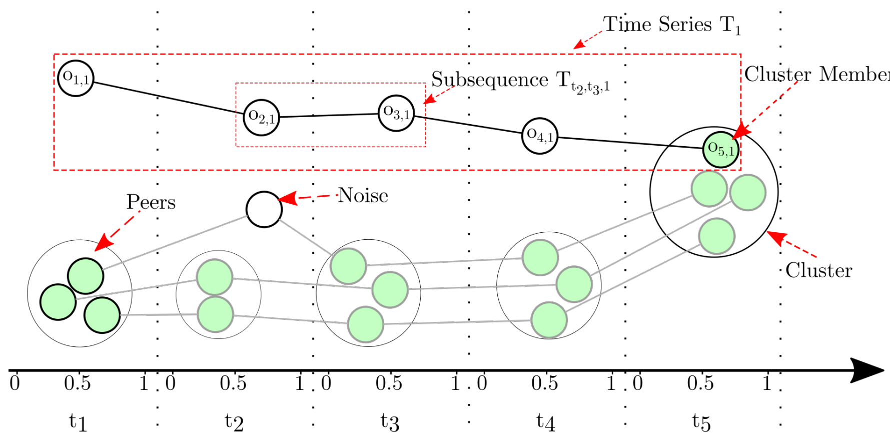


Fig. 1: Illustration of relevant terms regarding the time from t_1 to t_5.

proportions of two clusters. For this purpose we make use of the jaccard index which led to good results in [12] as well. Both extensions are tested separately and in conjunction. We highlight the differences and allow the user to choose carefully between those extensions - depending on his application.

Our original approach focuses on data sets with multivariate time series with discrete values, same length and equivalent time steps. Those time series are clustered per point in time and anomalous subsequences are detected by analysing the behavior of those. The behavior is defined as the change of peers over time. This leads us to the *subsequence_score* which represents the stability of a subsequence over time. An illustration of the relevant terms can be seen in Figure 1. With a calculated *outlier_score* for every subsequence and a threshold parameter τ we managed to detect anomalous time series. The outlier score depends on the subsequence score of a subsequence and the best subsequence score of a subsequence in the according cluster. In our work we differentiate three different types of outliers: *anomalous subsequences*, *intuitive outliers* and *noise*.

Our approach is different to other proposals which use cluster algorithms, as those either cluster time series as a whole [9] [11] [17], extract feature sets first [22], or consider subsequences of a single time series only [4]. None of the presented methods consider the cohesion of a time series regarding its peers. Our algorithm also differs from approaches which do not take time into account like [2] or which only regard subsequences of single time series [5] [15]. In contrary to those methods, we assume an information gain for one sequence from other sequences which have a semantic correlation.

In this paper we show once again, that we can identify an impact of other time series to one time series that is different to the granger causality [10] and that this influence can be used to detect anomalous subsequences. The adaptations of our original algorithm are well motivated and lead to different but sound results.

2 Related Work

Algorithms which detect outliers in time series are no novelty. There are actually various specialized approaches for different applications. Most methods deal with one time series only, while fewer ones regard multiple time series at the same time. There are different types of outliers, such as significantly deviating data points, uncommon subsequence patterns in periodic time series or changing points, which indicate that the further course of the sequence will change.
In many cases outliers of any type are identified with adapted autoregressive-moving-average (ARMA) models [3] [16]. Although these techniques are performing very well in most cases and factually are state-of-the-art, they lack the implementation of exterior information like other semantic correlated time series. There are also other methods which make use of decomposition techniques such as STL [6]. These methods work on time series which can be actually decomposed, but fail if this is not the case. Finally there are presented works which use dynamic time warping (DTW) [18] in order to detect anomalies.
There are also approaches which tackle the problem of finding outliers in multiple time series. Similar to our algorithm these methods are using peers of a time series to determine whether it is anomalous or not. The most recent works use Probabilistic Suffix Trees (PST) [19] or Random Block Coordinate Descents (RBCD) [23] in order to detect suspicious time series or subsequences. In contrary to our approach, in which the behavior of a time series is the central idea, the named methods analyse the deviation of one time series to the others. Our assumption that the change or the adherence of a time series to its peers is a crucial difference to all present methods. This behavior centered view is implemented by clustering time series per timestamp which is similar to identifying its peers per point in time. Then the movement of this time series relative to its peers is analysed. The result of this is described as a subsequence score, which also can be viewed as the stability over time of a time series regarding the adherence to its peers. The degree of change, also called transition, is an important factor to the subsequence score. It is also essential in cluster evolution methods such as [12], which try to match clusters of different time points. Works of this kind usually introduce a parameter which determines whether the dissimilarity of two clusters is too big to match. However, a match of clusters is a very subjective task and highly dependent on the used definitions. Further this is not necessary in order to detect outliers and thus not relevant for our work. The approach of Landauer et al. [14] uses an anomaly score, which is based on transitions of a single time series. This is different to our method, since we use the information of multiple time series.
The analysis of time series behavior like presented in this paper not only detects surprisingly deviating data points and subsequences with regard to a single time series, but also identifies new, behavior-based outliers. Our approach is also different from those which cluster whole time series, since such approaches do not consider the cluster transitions, which is an expressive feature on its own. The algorithm presented in this paper is able to detect anomalous subsequences, although they would have been assigned to one cluster in a subsequence clustering.

3 Fundamentals

Before introducing the method, some basic definitions regarding time series analysis used in the underlying paper [20] and this work are given, since they may vary in literature. An illustration of them can be seen in Figure 1.

Definition 1 (Time Series). *A multivariate time series* $T = o_{t_1}, ..., o_{t_n}$ *is an ordered set of* n *real valued data points of arbitrary dimension. The data points are chronologically ordered by their time of recording, with* t_1 *and* t_n *indicating the first and the last timestamp, respectively.*

Definition 2 (Data Set). *A data set* $D = T_1, ..., T_m$ *is a set of* m *time series of same length and equivalent points in time. The set of data points of all time series at a timestamp* t_i *is denoted as* O_{t_i}*.*

Definition 3 (Subsequence). *A subsequence* $T_{t_i,t_j,l} = o_{t_i,l}, ..., o_{t_j,l}$ *with* $j > i$ *is an ordered set of successive real valued data points beginning at time* t_i *and ending at* t_j *from time series* T_l*.*

Definition 4 (Cluster). *A cluster* $C_{t_i,j} \subseteq O_{t_i}$ *at time* t_i*, with* $j \in \{1, ..., q\}$ *being a unique identifier (e.g. counter) and* q *being the number of clusters, is a set of similar data points, identified by a cluster algorithm or human. This means that all clusters have distinct labels regardless of time.*

Definition 5 (Cluster Member). *A data point* $o_{t_i,l}$ *from time series* T_l *at time* t_i*, that is assigned to a cluster* $C_{t_i,j}$ *is called a member of cluster* $C_{t_i,j}$*.*

Definition 6 (Noise). *A data point* $o_{t_i,l}$ *from time series* T_l *at time* t_i *is considered as noise, if it is not assigned to any cluster.*

Definition 7 (Clustering). *A clustering is the overall result of a clustering algorithm or the set of all clusters annotated by a human for all timestamps. In concrete it is the set* $\zeta = \{C_{t_1,1}, ..., C_{t_n,q}\} \cup Noise$*.*

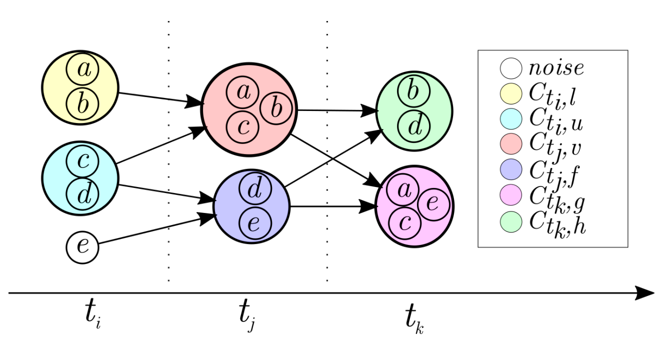


Fig. 2: Example for cluster transitions of time series $T_a, .., T_e$ over time.

An example for the above definitions can also be seen in Figure 2. Five time series of a data set $D = T_a$, T_b, T_c, T_d, T_e are clustered per timestamp for the time points t_i, t_j and t_k. The data points of a time series T_l are denoted by the identifier l for simplicity reasons. The shown clustering consists of six clusters. It can be described by the set $\zeta = \{C_{t_i,l}, C_{t_i,u}, C_{t_j,v}, C_{t_j,f}, C_{t_k,g}, C_{t_k,h}\} \cup \{o_{t_i,e}\}$. As $o_{t_i,e}$ is not assigned to any cluster in t_i, it is marked as noise for this timestamp. The data points $o_{t_i,a}, o_{t_i,b}$ of time series T_a and T_b in t_i are cluster members of the yellow cluster $C_{t_i,l}$.

4 Method

The cohesion of a sequence with its peers over time is described by the term *over-time stability*. Our approach is based on the assumption that unstable behavior over time indicates an irregularity. In order to rate the over-time stability of a sequence by means of a so called *subsequence_score*, the proportion of cluster members from earlier timestamps who migrated together into another cluster in later timestamps has to be calculated. For this reason, the *temporal cluster intersection* was introduced [20]:

$$\cap_t \{C_{t_i,a}, C_{t_j,b}\} = \{T_l \mid o_{t_i,l} \in C_{t_i,a} \land o_{t_j,l} \in C_{t_j,b}\}$$

with $C_{t_i,a}$ and $C_{t_j,b}$ being two clusters, $t_i, t_j \in \{t_1, ...t_n\}$ and $l \in \{1, ..., m\}$. The proportion p of two Clusters $C_{t_i,a}$ and $C_{t_j,b}$ with $t_i < t_j$ is then calculated by:

$$p(C_{t_i,a}, C_{t_j,b}) = \begin{cases} 0 & \text{if } C_{t_i,a} = \emptyset \\ \frac{|C_{t_i,a} \cap_t C_{t_j,b}|}{|C_{t_i,a}|} & \text{else} \end{cases}$$

As this proportion is asymmetric since it only describes the proportion of $C_{t_i,a}$ that is contained in $C_{t_j,b}$, a merge of clusters has no negative impact on the score. However, in some use cases it might be wanted to treat merges and splits equally, because a well-separated clustering is desired. With this calculation it is not possible to distinguish whether a time series has the best possible score because it always remains in its well-separated cluster or because its cluster only merged into other ones but never split off.

In order to punish merges and splits the same way, the jaccard index can be used to obtain the proportion. For this, we introduce the *temporal cluster union* of two clusters $C_{t_i,a}, C_{t_j,b}$:

$$\cup_t \{C_{t_i,a}, C_{t_j,b}\} = \{T_l \mid o_{t_i,l} \in C_{t_i,a} \lor o_{t_j,l} \in C_{t_j,b}\}$$

with $l \in \{1, ..., m\}$. Now the proportion $\hat{p}$ can be calculated by the jaccard index of two clusters:

$$\hat{p}(C_{t_i,a}, C_{t_j,b}) = \begin{cases} 0 & \text{if } C_{t_i,a} = \emptyset \land C_{t_j,b} = \emptyset \\ \frac{|C_{t_i,a} \cap_t C_{t_j,b}|}{|C_{t_i,a} \cup_t C_{t_j,b}|} & \text{else} \end{cases}$$

with $t_i < t_j$.
Regarding the example in Figure 2, the proportion p of cluster $C_{t_i,l}$ and $C_{t_j,v}$ would be

$$p(C_{t_i,l}, C_{t_j,v}) = \frac{|C_{t_i,l} \cap_t C_{t_j,v}|}{|C_{t_i,l}|} = \frac{2}{2} = 1$$

and therefore ideal. In contrast to that, the proportion $\hat{p}$ would be

$$\hat{p}(C_{t_i,l}, C_{t_j,v}) = \frac{|C_{t_i,l} \cap_t C_{t_j,v}|}{|C_{t_i,l} \cup_t C_{t_j,v}|} = \frac{2}{3} = 0.\overline{6}$$

as the merge of cluster $C_{t_i,l}$ and $C_{t_i,u}$ lowers the score.
Using the proportion, each subsequence $T_{t_i,t_j,l}$ of time series l beginning at timestamp t_i and ending at t_j is rated by the following *subsequence_score* in [20]:

$$subsequence\_score(T_{t_i,t_j,l}) = \frac{1}{k} \cdot \sum_{v=i}^{j-1} p(cid(o_{t_v,l}), cid(o_{t_j,l}))$$

with $l \in \{1, ..., m\}$, $k \in [1, j-i]$ being the number of timestamps between t_i and t_j where the data point exists and *cid*, the cluster-identity function

$$cid(o_{t_i,l}) = \begin{cases} \emptyset & \text{if the data point is not assigned to any cluster} \\ C_{t_i,a} & \text{else} \end{cases}$$

returning the cluster which the data point has been assigned to in t_i. In words, it is the average proportion of the sequence's clusters it migrated with from t_i to t_j. Here, the impact of all preceding time points to the score is weighted equally. For longer sequences, this can lead to a tendency towards a worse rating, since slow changes in cluster membership might influence the rating quite considerably. Assuming that the nearer past is more meaningful than the more distant past, we formulate a weighting that can be used in the subsequence score.
Regarding a time interval $[t_1, t_k]$, the proportion at time t_i with $t_1 \leq t_i \leq t_k$ gets the weighting $\frac{2 \cdot i}{k(k+1)}$ resulting by the division of i with the Gauss's Formula

$$\frac{i}{\sum_{a=1}^{k} a} = \frac{i}{\frac{k(k+1)}{2}} = \frac{2 \cdot i}{k(k+1)} \ .$$

The weighting function can easily be adjusted for time intervals starting at time $t_s > t_1$. The subsequence score is then calculated as follows:

$$weighted\_subseq\_score(T_{t_i,t_j,l}) = \sum_{v=i}^{j-1} \frac{2 \cdot (v-i+1)}{k(k+1)} p(cid(o_{t_v,l}), cid(o_{t_j,l}))$$

with $k \in [1, j-i]$ again being the number of timestamps between t_i and t_j where the data point exists. Since the sum of all weightings of a subsequence's timestamps is always 1, there is no need to normalize the score to an interval of

$[0, 1]$ by averaging it.
In the example of Figure 2, the score of time series T_a between time points t_i and t_k would be

$$subsequence\_score(T_{t_i,t_k,a}) = \frac{1}{2} \cdot (1.0 + 0.\overline{6}) = 0.8\overline{3}$$

whereby the rating with the weighted subsequence score would be

$$weighted\_subseq\_score(T_{t_i,t_k,a}) = (\frac{1}{3} \cdot 1.0 + \frac{2}{3} \cdot 0.\overline{6}) = 0.78$$

The second proportion which is smaller than 1 has thus more influence on the score now. The combination of the weighted subsequence score and the jaccard proportion $\hat{p}$ has the following result:

$$weighted\_jaccard\_score(T_{t_i,t_k,a}) = (\frac{1}{3} \cdot 0.\overline{6} + \frac{2}{3} \cdot 0.5) = 0.56$$

With the help of the subsequence's rating an outlier score can be calculated for each by determining the deviation of their stability from the best subsequence score of their cluster. Formally, the best score of a cluster $C_{t_j,a}$ for sequences starting at t_i and ending at t_j is given by

$$best\_score(t_i, C_{t_j,a}) = max(\{subsequence\_score(T_{t_i,t_j,l}) \mid cid(o_{t_j,l}) = C_{t_j,a}\}) \ .$$

A subsequence's outlier score is then described by

$$outlier\_score(T_{t_i,t_j,l}) = best\_score(t_i, cid(o_{t_j,l})) - subsequence\_score(T_{t_i,t_j,l}) \ .$$

The outlier score is therefore dependent on the over-time stability of the considered cluster's members. The smaller the best score is, the smaller is the highest possible outlier score. The detection of outlier sequences can be done by using a threshold τ [20]:

Definition 8 (Outlier). *Given a threshold $\tau \in [0, 1]$, a subsequence $T_{t_i,t_j,l}$ is called an outlier, if its probability of being an outlier is greater than or equal τ. That means, if*

$$outlier\_score(T_{t_i,t_j,l}) \geq \tau \ .$$

In addition to these outlier sequences, subsequences that consist entirely of noise data points from the clustering algorithm are identified as *intuitive outliers*. Sequences whose last data point is labeled as noise are not assigned to a cluster which the best score can be determined from, so they do not get an outlier score.

5 Experiments

In the following, several experiments on different (artificially generated and real world) data sets are performed in order to evaluate the effects of the modifications of this paper regarding the original method. In all cases the density-based clustering algorithm *DBSCAN* [8] was used for clustering. We will differentiate between the *original method* from [20], the *jaccard method* (where the proportion is calculated by the jaccard index), the *weighted method* (where the weighting is included in the subsequence score), and the *weighted jaccard method* (where all modifications are integrated). In all experiments the same parameter settings for ϵ, $mitPts$ and τ were used for the investigated methods in order to make the results comparable. Please note, that dependent on the method in some cases another parameter choice could have been beneficial.

5.1 Artificially Generated Data Set

For a targeted evaluation of the properties, at first an artificially generated data set with 40 timestamps is considered. The data set was generated so that initially four starting points (for four groups of time series) were selected. In addition, the maximum distance of the centroids of two successive time points and the number of members were chosen for each group. The centroids as well as the members' data points were then calculated randomly for each time point, whereby the distance of the members to the centroids could not exceed 0.03. After generating the normal data points, one completely random outlier sequence and three targeted outlier sequences were inserted. For the completely random sequence all data points were chosen randomly and the distance between two consecutive points was set to not being greater than 0.1. The remaining outlier sequences were generated as follows: The data points were always set with a maximum distance of 0.06 to a centroid. The clusters were chosen randomly whereby the distance of the latest data point and the next centroid could not exceed 0.2. Additionally, the sequence always had to be allocated for at least 5

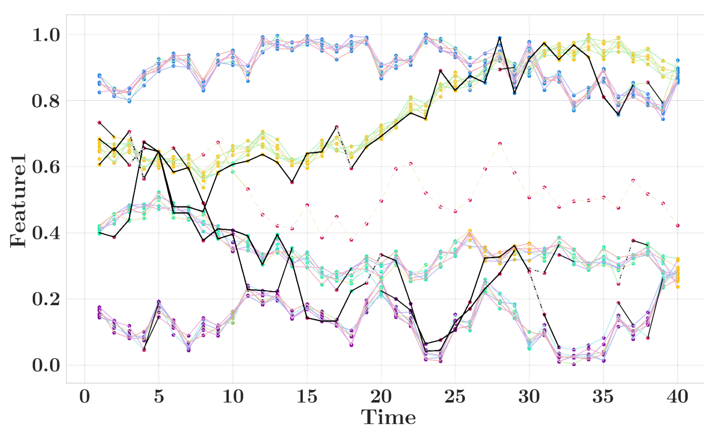


Fig. 3: Achieved results on the generated data set with $\epsilon = 0.025$, $minPts = 3$ and $\tau = 0.7$ by the original method.

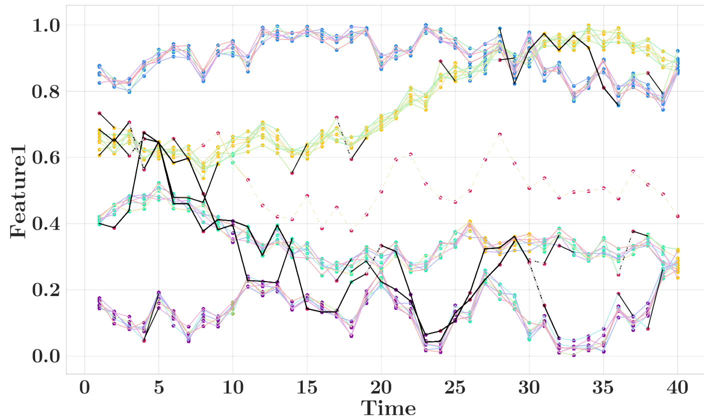


Fig. 4: Achieved results on the generated data set with $\epsilon = 0.025$, $minPts = 3$ and $\tau = 0.7$ by the weighted method.

time points to the same cluster before choosing the next one. For all points, care was taken to ensure that they were between 0 and 1.

The time series data was clustered per timestamp with the parameter setting $\epsilon = 0.025$ and $minPts = 3$. All four methods were performed on the clustering with the threshold $\tau = 0.7$. The results are illustrated in the Figures Fig. 3, Fig. 4, Fig. 5 and Fig. 6. Red dots represent noise data points while other colors indicate the cluster membership. Black lines stand for outliers that are found with the outlier score and dashed lines represent intuitive outliers.

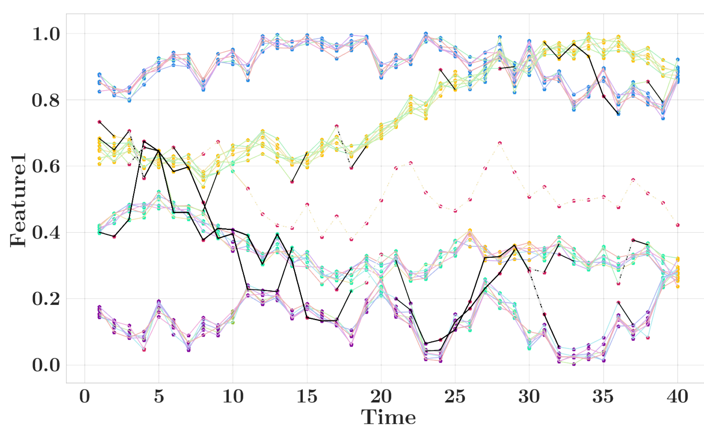


Fig. 5: Achieved results on the generated data set with $\epsilon = 0.025$, $minPts = 3$ and $\tau = 0.7$ by the jaccard method.

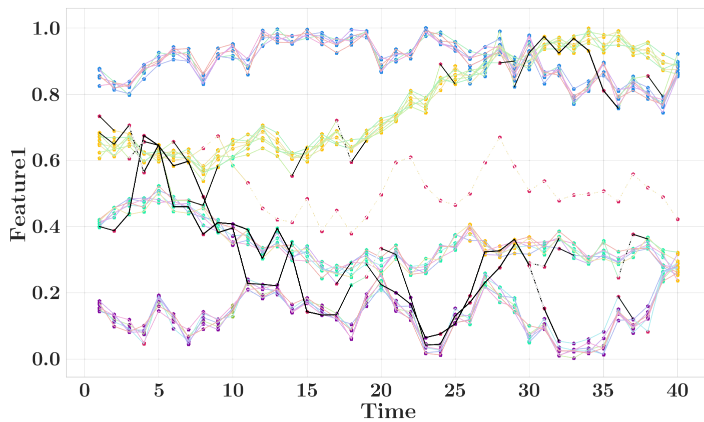
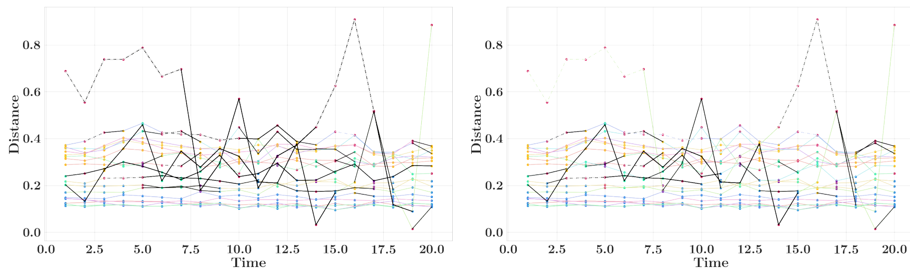


Fig. 6: Achieved results on the generated data set with $\epsilon = 0.025$, $minPts = 3$ and $\tau = 0.7$ by the weighted jaccard method.

The original method (Fig. 3) detects all four outlier sequences and marks almost the whole time series as such. However, some parts of the outlier sequence in the yellow clusters (second from the top) are quite stable and therefore should not be detected as outliers in regard to their over-time stability. When considering the results of the weighted method (Fig. 4) one can see, that some smaller parts of the time series are marked as outliers. The most obvious example is the outlier sequence of the yellow clusters. This effect shows, that the intention of the weighting, that the more distant past has a lower impact on the score than the nearer past, is therefore satisfied. The jaccard method (Fig. 5) leads to a more sparsely detection, as well. This can be explained by the fact that due to some merges (for example in the yellow clusters) the best subsequence score of the clusters is decreased and consequently the highest outlier score is decreased, too. The effect of the lower best score can also be seen between the timestamps 29 and 35. In contrast to the weighted method, the "M" shape is not marked completely. The combination of both modifications is illustrated in Figure 6. Since the nearer past is weighted more strongly here, the merge of the blue and yellow clusters at time point 26 has not as much influence on the best score. Therefore the "M" shape is detected as outlier. However, there are some differences in regard to the results of the weighted method. Overall fewer outlier sequences are found. An example can be seen in the first time stamps. This behavior is reasoned as the jaccard index lowers the best possible score in the clusters.

5.2 Airline On-Time Performance Data Set

This data set holds 29 features like the scheduled and actual departure time for flights reported by certified U.S. air carriers. In total it contains 3.5 million



(a) Original Method (b) Weighted Jaccard Method

Fig. 7: Achieved results on the Airline On-Time Performance Data Set with $\epsilon = 0.03$, $minPts = 3$ and $\tau = 0.5$.

records with each representing a flight. Originally this data set is provided by the U.S. Department of Transportation's Bureau of Transportation Statistics [7]. In order to make the data set suitable for our approach we interpreted the feature set of every airline as a sequence. Further we made these time series equidistant by calculating the average of their features for every day. Finally we normalized the data set with the min-max normalization and clustered it per timestamp.
In this experiment we compare the original method [20] with the modified approach presented in this paper. Both modifications are applied and the result is illustrated in Figure 7b. The first noticeable difference to the original [20] approach in Figure 7a is the lower amount of marked outliers. This can be explained with both adjustments: First of all, the introduced jaccard index leads to overall lower subsequence scores, thus the best score of a cluster is lower and therefore the outlier score is lower. Second, the weighting function allows time series to change their peers over time if it is done consequently. This means that time series are not considered to be suspicious if they made a stable change, which is to expect when regarding larger time series. Actually the original approach cannot handle the amount of points in time and tends to become more sensitive with rising amount of time stamps. In contrary, the adjusted version performs more robust and can handle more timestamps better.
On the second sight, one might notice that the adjusted method detects slightly different outliers than the original approach (e.g. the two upper outliers between timestamp 17.5 and 20.0). However, those differences in this example are too small to be reasoned with a specific modification.

5.3 GlobalEconomy Data Set

The GlobalEconomy data set is obtained from the website theglobaleconomy.com [1]. It holds over 300 indicators for different features for 200 countries over more than 60 years. For illustration reasons we chose 20 countries and two features, namely the education spendings and the unemployment rate. Please note, that the amount of countries can vary per timestamp, because there are missing values in the data set.

The result of the original method and the modified approach presented in this paper, can be seen in Figure 8 and Figure 9. The colors represent the detected clusters, circles represent behavior-based outliers and red font is indicating noise which was detected by DBSCAN. In case a country is detected as a behavior-based outlier and as noise by DBSCAN it is represented as a circle with red font. The abbreviations are according to ISO 3166. At first glance it is noticeable that our original approach detected more outliers than the new method. Let us explain this by the example of Kyrgyzstan (KGZ) in the years 2010 and 2011: KGZ leaves the yellow cluster and at the same time joins the green cluster in 2011. In our original calculation KGZ is punished for this transition by applying the old cluster proportion function. At the same time the subsequence score of the Marshall Islands (ISL) is not influenced in 2011, because it was not assigned to a cluster in 2010. Thus the outlier score of Kyrgyzstan is negatively influenced. In the weighted jaccard method Kyrgyzstan is not detected as an outlier, because the Marshall Islands are punished for the merge with Kyrgyzstan in 2011. This leads to a lower *best_score* and at the same time to a lower *outlier_score* of Kyrgyzstan. In summary, Kyrgyzstan is not detected as an outlier, because the Marshall Islands are now punished for merging.

An example of finding new outliers is Honduras (HND) in the years from 2013 to 2015. The old technique did not identify Honduras as an outlier in the years 2014 and 2015, while the modified method does. Again this has to do with the low subsequence score of the Marshal Islands in 2014, but this time the cluster proportion of the original approach is punishing the Marshal Islands for splitting

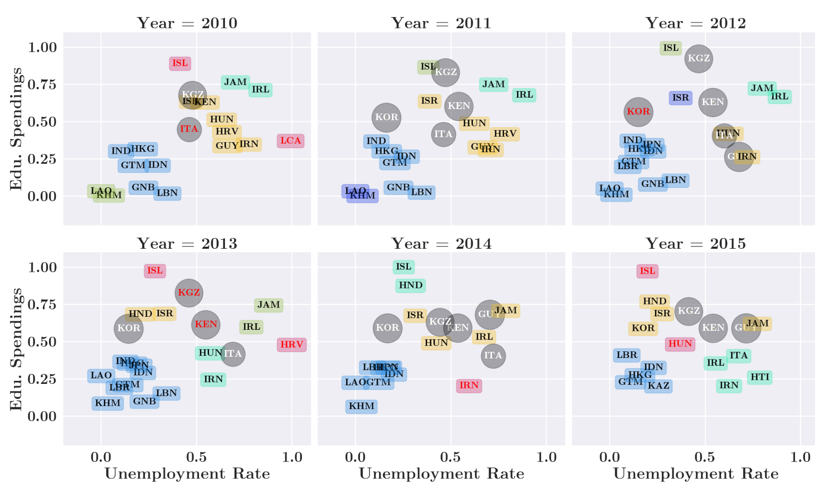


Fig. 8: Achieved results on the GlobalEconomy Data Set with $\epsilon = 0.18$, $minPts = 2$ and $\tau = 0.4$ by the original method.

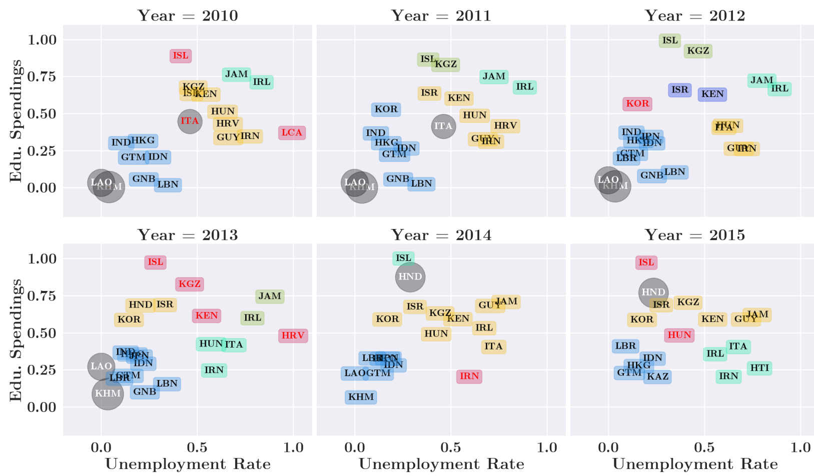


Fig. 9: Achieved results on the GlobalEconomy Data Set with $\epsilon = 0.18$, $minPts = 2$ and $\tau = 0.4$ by the weighted jaccard method.

from its peers in the previous years. However, this is not the only reason Honduras is not marked as an outlier. It actually benefits from rejoining the yellow cluster in 2015, although the yellow cluster contains more than twice the amount of countries now. In concrete that means, that the comparison of the years of 2013 and 2015 was influencing the subsequence score of Honduras in a positive way. The weighted jaccard approach takes the new constellation of the yellow cluster into account. In contrary to the original method, the comparison of the years 2013 and 2015 is not beneficial to its subsequence score. Further more the merge with the Marshall Islands in 2014 is punished by the jaccard index.
Another interesting observation is, that the jaccard index now enables the identification of outlier clusters. In Figure 9 one can observe, that Laos (LAO) and Cambodia (KHM) form a cluster in the years from 2010 to 2011. The merge to the big blue cluster in 2012 has a fairly high influence on their subsequence scores so that they are detected as outliers in 2013. Although the two countries are more stable in the subsequence from 2012 to 2013, they have not stabilized to the level of their peers in the blue cluster. This finally happens in the year of 2014.

6 Conclusion & Future Work

The analysis of time series data – especially the identification of conspicuous sequences – is an important field in data mining. So far, there are only a few approaches for the detection of outliers in multiple time series. In [20] we presented an outlier detection algorithm which analyses the behavior of groups of time series by clustering the data per timestamp using an arbitrary clustering

algorithm. As this was a novel approach, there were still some handicaps and application dependent properties. In this paper, we focused on two of these and proposed the following solutions: First, we presented another technique for the calculation of the proportion, which treats merges and splits of clusters equally. Second, we introduced a weighting function that causes a higher impact of a sequence's nearer past than the more distant one. Our results show, that the intended effects were achieved by our modifications. All results are meaningful and show individual qualities. Dependent on the application, one of the four investigated methods can be used for the detection of anomalous subsequences in regard to their over-time stability.
However, the aspects dealt with in this paper were only a part of the procedure's difficulties. There is still the problem of determining the best parameter τ and optimal hyperparameters for the clustering algorithms such as DBSCAN. Additionally, the treatment of noise data points could be improved. As proposed in [20], the inclusion of the time series' deviations might lead to an advanced analysis of those. Further, the detection of outlier clusters would be interesting. Partly they are already found by the modified method presented in this paper. Finally, the procedure could be adjusted to handle fuzzy clusterings. With the help of over-time stability measures for hard [21] and fuzzy clusterings [13] a good basis for the outlier detection can be provided.

7 Acknowledgement

We would like to thank the Jürgen Manchot Foundation, which supported this work by financing the AI research group *Decision-making with the help of Artificial Intelligence* at Heinrich Heine University Düsseldorf.

References

1. Global economy, world economy, https://www.theglobaleconomy.com/.
2. Ahmad, S., Lavin, A., Purdy, S., Agha, Z.: Unsupervised real-time anomaly detection for streaming data. Neurocomputing **262**, 134 – 147 (2017)
3. Ahmar, A.S., Guritno, S., Abdurakhman, Rahman, A., Awi, Alimuddin, Minggi, I., Tiro, M.A., Aidid, M.K., Annas, S., Sutiksno, D.U., Ahmar, D.S., Ahmar, K.H., Ahmar, A.A., Zaki, A., Abdullah, D., Rahim, R., Nurdiyanto, H., Hidayat, R., Napitupulu, D., Simarmata, J., Kurniasih, N., Abdillah, L.A., Pranolo, A., Haviluddin, Albra, W., Arifin, A.N.M.: Modeling data containing outliers using ARIMA additive outlier (ARIMA-AO). Journal of Physics: Conference Series **954** (2018)
4. Banerjee, A., Ghosh, J.: Clickstream clustering using weighted longest common subsequences. In: In Proceedings of the Web Mining Workshop at the 1st SIAM Conference on Data Mining. pp. 33–40 (2001)
5. Cheng, H., Tan, P.N., Potter, C., Klooster, S.: Detection and characterization of anomalies in multivariate time series. In: Proceedings of the 2009 SIAM international conference on data mining. pp. 413–424 (2009)
6. Cleveland, R.B., Cleveland, W.S., McRae, J.E., Terpenning, I.: Stl: A seasonal-trend decomposition procedure based on loess (with discussion). Journal of Official Statistics **6**, 3–73 (1990)

7. Computing, S., Graphics, S.: Airline on-time performance. http://stat-computing.org/dataexpo/2009/the-data.html, accessed: 2019-07-15
8. Ester, M., Kriegel, H.P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. pp. 226–231 (1996)
9. Ferreira, L.N., Zhao, L.: Time series clustering via community detection in networks. Information Sciences **326**, 227 – 242 (2016)
10. Granger, C.W.J.: Investigating causal relations by econometric models and cross-spectral methods. Econometrica **37**(3), 424 (Aug 1969)
11. 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)
12. Kalnis, P., Mamoulis, N., Bakiras, S.: On discovering moving clusters in spatio-temporal data. In: Bauzer Medeiros, C., Egenhofer, M.J., Bertino, E. (eds.) Advances in Spatial and Temporal Databases. pp. 364–381 (2005)
13. Klassen, G., Tatusch, M., Himmelspach, L., Conrad, S.: Fuzzy clustering stability evaluation of time series. In: Information Processing and Management of Uncertainty in Knowledge-Based Systems. pp. 680–692 (2020)
14. Landauer, M., Wurzenberger, M., Skopik, F., Settanni, G., Filzmoser, P.: Time series analysis: Unsupervised anomaly detection beyond outlier detection. In: ISPEC (2018)
15. Malhotra, P., Vig, L., Shroff, G.M., Agarwal, P.: Long short term memory networks for anomaly detection in time series. In: ESANN (2015)
16. Munir, M., Siddiqui, S.A., Chattha, M.A., Dengel, A., Ahmed, S.: Fusead: Unsupervised anomaly detection in streaming sensors data by fusing statistical and deep learning models. Sensors **19**(11) (2019)
17. Paparrizos, J., Gravano, L.: k-shape: Efficient and accurate clustering of time series. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data. pp. 1855–1870 (2015)
18. Salvador, S., Chan, P.: Toward accurate dynamic time warping in linear time and space. Intell. Data Anal. **11**(5), 561–580 (2007)
19. Sun, P., Chawla, S., Arunasalam, B.: Mining for outliers in sequential databases. In: in ICDM, 2006. pp. 94–106
20. Tatusch, M., Klassen, G., Bravidor, M., Conrad, S.: Show me your friends and i'll tell you who you are. finding anomalous time series by conspicuous cluster transitions. In: Data Mining. AusDM 2019. Communications in Computer and Information Science. vol. 1127, pp. 91–103 (2019)
21. Tatusch, M., Klassen, G., Bravidor, M., Conrad, S.: How is your team spirit? cluster over-time stability evaluation (forthcoming). In: Machine Learning and Data Mining in Pattern Recognition, 16th International Conference on Machine Learning and Data Mining, MLDM (2020)
22. Truong, C.D., Anh, D.T.: A novel clustering-based method for time series motif discovery under time warping measure. International Journal of Data Science and Analytics **4**(2), 113–126 (2017)
23. Zhou, Y., Zou, H., Arghandeh, R., Gu, W., Spanos, C.J.: Non-parametric outliers detection in multiple time series a case study: Power grid data analysis. In: AAAI (2018)