Robust Division in Clustering of Streaming Time Series

Pedro Pereira Rodrigues\(^1\) and João Gama\(^2\)

Abstract. Online learning algorithms which address fast data streams should process examples at the rate they arrive, using a single scan of data and fixed memory, maintaining a decision model at any time and being able to adapt the model to the most recent data. These features yield the necessity of using approximate models. One problem that usually arises with approximate models is the definition of a minimum number of observations necessary to assure convergence, which implies a high risk since the system may have to decide based only on a small subset of the entire data. One approach is to apply techniques based on the Hoeffding bound to enforce decisions with a confidence level. In divisive clustering of time series, the goal is to find clusters of similar time series over time. In online approaches there are two decisions to make: when to split and how to assign variables to new clusters. We can define a confidence level to both the decision of splitting and the assignment of data variables to new clusters. Previous works have already addressed confident decisions on the moment of split. Our proposal is to include a confidence level to the assignment process. When a split point is reported, creating two new clusters, we can directly assign points which are confidently closer to one cluster than the other, having a different strategy for those variables which do not satisfy the confidence level. In this paper we propose to assign the unsure variables to a third cluster. Experimental evaluation is presented in the context of a recently proposed hierarchical algorithm, assessing the advantages of our proposal, while Section 6 presents some concluding remarks.

Keywords: robust clustering, time series, data streams.

1 INTRODUCTION

Data streams usually consist of variables producing examples continuously over time. Clustering streaming time series is an emergent field in data mining and knowledge discovery from data streams. The basic idea behind this task is to find groups of variables (the time series) that behave similarly through time, which is usually measured in terms of time series similarities. Several real-world clustering problems exist that address data coming from a stream at high rate: in electrical supply systems, clustering demand profiles (ex: industrial or urban) decreases the computational cost of predicting each individual subnetwork load [5]; in medical systems, clustering medical sensor data (such as ECG, EEG, etc.) is useful to determine correlation between signals [14]; in financial markets, clustering stock prices evolution helps preventing bankruptcy [10]. Hence, data stream approaches should be considered to solve them.

In this work we address the problem of clustering streaming series assuming data is gathered by a centralized process while it is becoming available for online analysis. Recent research as addressed this issue from different perspectives. Beringer and Hüllemeyer proposed an online version of k-means for clustering parallel data streams, using a Discrete Fourier Transform approximation of the original data [1]. The basic idea is that the cluster centers computed at a given time are the initial cluster centers for the next iteration of k-means, applying a procedure to dynamically update the optimal number of clusters at each iteration. Clustering On Demand (COD) is another framework for clustering streaming series which performs one data scan for online statistics collection and has compact multi-resolution approximations, designed to address the time and the space constraints in a data stream environment [2]. It is divided in two phases: a first online maintenance phase providing an efficient algorithm to maintain summary hierarchies of the data streams and retrieve approximations of the sub-streams; and an offline clustering phase to define clustering structures of multiple streams with adaptive window sizes. Rodrigues et al. [13], proposed the Online Divisive-Agglomerative Clustering (ODAC) system, a hierarchical procedure which dynamically expands and contracts clusters based on their diameters. It constructs a tree-like hierarchy of clusters of streams, using a top-down strategy based on the correlation between streams.

In the next section we present the main contribution of this paper, a robust criterion for the assignment of time series to new clusters. Since we decided to evaluate the criterion in the context of the ODAC system, an overview on this recent hierarchical algorithm is included in section 3, exposing its main characteristics. Section 4 enunciates the validity indices used in Section 5 to validate and support the advantages of our proposal, while Section 6 presents some concluding remarks.

2 A ROBUST ASSIGNMENT CRITERION

In divisive clustering algorithms, when a split point is reported, the different systems take different actions. For example, ODAC [13] determines two variables as pivots and assigns each of the remaining variables to the cluster which has the closest pivot. This is usually a good heuristic, as it often finds an optimal border hyperplane. It is a lot faster than the heuristic performed by DIANA [9], since it does not need to compute the average distances to decide which leaf will receive each variable. However, this may lead to erroneous situations if the moving variable is equally distant from the two pivots, as there is no way of determining to which cluster it should be assigned.

This symmetric assignment is the key object of our proposal in this work. When considering the expansion of the structure, the symmetric splitting of variables appears as a possible drawback, in the

---

\(1\) LIAAD - INESC Porto L.A. & Faculty of Sciences, University of Porto, Portugal, email: pprodrigues@fc.up.pt

\(2\) LIAAD - INESC Porto L.A. & Faculty of Economics, University of Porto, Portugal, email: jgama@fep.up.pt
sense that a previous decision of moving a variable to a leaf, when there was no statistical confidence on the decision of assignment, may separate variables that should be together. The main issue here is the possible splitting of compact clusters due to their equidistant position to two variables external to that cluster. Left plot of figure 1 presents an example of a possible configuration where the problem could arise. If we could assign a confidence level to the assignment, only variables which were confidently closer to one pivot than the other would follow the corresponding pivot.

2.1 Assignment alternatives

The problem of assigning evenly dissimilar variables has two possible approaches. On one side, we can simply disregard the problem and assign them to the closest pivot. On the other side, we can include a confidence to the decision of assignment, directly assigning variables which are confidently closer to one pivot than the other, and having a different strategy for those variables which do not verify the confidence level, further called **unsure** variables.

The Hoeffding bound [8] can be used to control this decision, having the advantage of being independent of the probability distribution generating the observations [3], and stating that after $n$ independent observations of a real-valued random variable $r$ with range $R$, and with confidence $1 - \delta$, the true mean of $r$ is at least $\mathcal{T} - \epsilon$, where $\mathcal{T}$ is the observed mean of the samples and

$$\epsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}}.$$  

(1)

As each leaf is fed with a different number of examples, each cluster $c_k$ will possess a different value for $\epsilon$, designated $\epsilon_k$.

Let $d_1 = d(u, p_1)$ and $d_2 = d(u, p_2)$ be the distance measure between the **unsure** time series $u$ and each of the chosen pivots $p_1$ and $p_2$. After seeing $n$ samples at the leaf, let $\Delta d = d_1 - d_2$, a new random variable consisting on the difference between the observed values through time. Applying the Hoeffding bound to $\Delta d$, if $\Delta d > \epsilon_k$, one can confidently say that, with probability $1 - \delta$, the difference between $d_1$ and $d_2$ is larger than zero, and assign $u$ to the cluster represented by $p_1$ (the equivalent for $p_2$ holds). For the remaining, several approaches exist:

- a **fuzzy** approach, where the **unsure** variables are assigned to both clusters, with a membership weight;
- a **robust** approach, where the **unsure** variables are assigned to a third cluster.

In a cluster with $n$ variables, speed is the main motivation of a symmetric assignment, being $O(n)$. The first approach, although it may prevent the symmetry issue, has the problem of reducing the ability to cope with high-speed data streams, as each assignment test has to compute the average dissimilarity of the variable to each cluster ($O(n^2)$).

The second approach presents several advantages on the quality of the resulting clusters, as the decision of assignment is delayed to lower levels of the hierarchy (also $O(n)$). This was recently proposed in [12], revealing both its benefits and its drawbacks. As reported, the downside of this is that, since one stream could appear in several clusters, the memory and processing advantages would loose its relevance.

The third approach combines the advantage of confident assignment with fast processing ($O(n)$), while keeping the memory requirements. Moreover, since some of the clusters are split in three, the reduction in memory size and speed-up in processing time with every split has even more impact on the global performance of the system.

2.2 Confident assignment strategy

Let $p_1$ and $p_2$ be the pivots of the clusters $c_1$ and $c_2$, respectively, and $u$ be the moving variable. The expansion is decided as follows:

- if $d(p_2, u) - d(p_1, u) > \epsilon_k$ move variable to cluster $c_1$;
- if $d(p_1, m) - d(p_2, m) > \epsilon_k$ move variable to cluster $c_2$;
- else move $m$ into a **third** cluster $c_3$, which will keep the variables that do not fall into any of the two groups.

The main advantage of this procedure is to prevent splitting compact clusters due to their symmetrical position between two variables external to that cluster. An example of application of this confident assignment is explained in the right plot of figure 1. In the proposed approach, the expansion still keeps the characteristic of speeding up the process with structure growth. Moreover, we should stress that the third sibling will only be created if at least one variable cannot be confidently assigned. If the data can be easily symmetrically bisected, only the two usual clusters are created.

This approach is a simple variation of the ODAC algorithm that might be useful in applications where the data reveals a multi-diversity characteristic which may not be easily structured in a binary hierarchy. Moreover, it is our belief that this approach would reach the real final clusters faster than the original procedure.

To assess the benefits of this new strategy, we evaluate the new criterion in the context of the ODAC clustering system. This way, next sections include a brief but necessary overview of this algorithm. We should stress that, although motivated by an existent method, this confident assignment criterion can easily be applied to any divisive procedure other than ODAC.

3 ODAC OVERVIEW

The Online Divisive-Agglomerative Clustering (ODAC) is an incremental approach for clustering streaming time series using a hierarchical procedure [13]. It constructs a tree-like binary hierarchy of...
clusters of streams, using a top-down strategy based on the correlation between streams. The system also possesses an agglomerative phase to enhance a dynamic behavior capable of structural change detection. The splitting and agglomerative operators are based on the diameters of existing clusters and supported by a significance level given by the Hoeffding bound [8]. Inspecting the algorithm, we can observe that:

- the update time and memory consumption does not depend on the number of examples, as it gathers sufficient statistics to compute the correlations within each cluster; moreover, anytime a split is reported, the system becomes faster as less correlations must be computed;
- the system possesses an anytime compact representation, since a binary hierarchy of clusters is available at each time stamp, and does not need to store anything more than the sufficient statistics;
- an agglomerative phase is included to react to structural changes; these changes are detected by monitoring the diameters of existing clusters;
- given its hierarchical core, the system possesses an inherently adaptable configuration of clusters.

This is one of the systems clearly proposed to address clustering of multiple streams. It copes with high-speed production of examples and reduced memory requirements, with constant update time. It also presents adaptability to new data, detecting and reacting to structural drift.

### 3.1 Dissimilarity measure

The system must analyze distances between incomplete vectors, possibly without having any of the previous values available. Thus, these distances must be incrementally computed. The system uses Pearson’s correlation coefficient [11] between time series as similarity measure. This way, the sufficient statistics needed to compute the correlation are easily updated at each time step.

### 3.2 Splitting criterion

One problem that usually arises with approximate models is the definition of a minimum number of observations necessary to assure convergence. One approach is to apply techniques based on the Hoeffding bound [8] to solve this problem.

Let \(d(a, b)\) be the distance measure between pairs of time series, and \(D_k = \{(x_i, x_j) \mid x_i, x_j \in c_k, i < j\}\) be the set of pairs of variables included in a specific leaf \(c_k\). After seeing \(n\) samples at the leaf, let

\[
(x_1, y_1) = \arg\max_{(x, y) \in D_k} d(x, y)
\]

be the pair of variables with maximum dissimilarity within the cluster \(c_k\), and in the same way considering

\[
D'_k = D_k \setminus \{(x_1, y_1)\},
\]

let

\[
(x_2, y_2) = \arg\max_{(x, y) \in D'_k} d(x, y)
\]

be the second top-most dissimilar pair of variables. Consider \(d_1 = d(x_1, y_1)\) and \(d_2 = d(x_2, y_2)\) in \(\Delta d = d_1 - d_2\), a new random variable consisting on the difference between the observed values through time. Applying the Hoeffding bound to \(\Delta d\), if \(\Delta d > \epsilon_k\), one can confidently say that, with probability \(1 - \delta\), the difference between \(d_1\) and \(d_2\) is larger than zero, and select \((x_1, y_1)\) as the pair of variables representing the diameter of the cluster.

With this rule, the ODAC system will only split the cluster when the true diameter of the cluster is known with statistical confidence given by the Hoeffding bound. However, to prevent the hierarchy from growing unnecessarily, another criterion is defined in ODAC which has to be fulfilled in order to perform the splitting, which falls out of the scope of this work.

### 3.3 Original assignment criterion

When a split point is reported, the pivots are variables \(x_1\) and \(y_1\) where \(d_1 = d(x_1, y_1)\), which are separated into each of the newly created clusters. The system then assigns each of the remaining variables of the old cluster to the cluster which has the closest pivot. The previous section introduced our proposal to solve the uncertainty of this assignment when dealing with streaming time series.

### 3.4 Aggregation criterion

The main setting of the system is the monitoring of existing clusters’ diameters. On stationary data streams, the diameter of a cluster decreases every time a split occurs. However, usual real-world problems deal with non-stationary data streams, where time series that were correlated in the past are no longer correlated to each other, in the current time period, and might be approaching time series of other clusters. The strategy that is adopted in ODAC to detect changes in the structure is based only on the analysis of the diameters. In fact, the diameter of each two new clusters should be less or equal than their parent’s diameter. In this way, no computation is needed between the variables of the two siblings.

### 4 CLUSTER VALIDITY

To validate our proposal, we will measure quality in a two-fold fashion: the quality of the hierarchy with respect to the data, and the correspondence between the final clustering structure and the real data. The Cophenetic Correlation Coefficient [7] (CPCC) measures quality in hierarchical structures, being defined as

\[
\text{CPCC} = \frac{1}{\sqrt{N \cdot N}} \left( \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} C_{ij} P_{ij} - \mu_P \mu_C \right)
\]

where

\[
\mu_P = \frac{1}{M} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} P_{ij}
\]

\[
\mu_C = \frac{1}{N^2} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} C_{ij}
\]

The closer the value of this index is to 1, the better the match and the better the hierarchy fits the data.

Regarding the correspondence between the clustering structure and the real data, two criteria give different insights on the result.
The Modified Hubert’s Γ Statistic index [6] is given by

\[ MHT\Gamma = \frac{1}{M} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} P_{i,j}Q_{i,j} \]  

(3)

where \( M = \frac{n(n-1)}{2} \), \( P \) is the proximity matrix and \( Q \) is a \( N \times N \) matrix where each \( Q_{i,j} \) is the distance between the representative points (centroids, medoids, etc.) of the clusters to which \( i \) and \( j \) belong. High values of this index represent compact and well-separated clusters. The Dunn’s Validity Index [4] is given by

\[ DVI = \min_{i,j} \left\{ \frac{d(c_i, c_j)}{\max\{diam(c_k)\}} \right\} \]  

(4)

where \( d(c_i, c_j) \) is the single linkage dissimilarity function between two clusters, and \( diam(c_k) \) is the diameter of cluster \( c_k \). High values of this index also represent compact and well-separated clusters. The MHT measure increases with the number of clusters, but the second index is independent of the number of clusters. This way, the highest value of DVI is considered the best one.

5 EXPERIMENTAL EVALUATION

The main argued advantage of this technique is to prevent splitting compact clusters due only to the fact that they might be equidistant to the two chosen pivots. Moreover, it is also our belief that this approach would reach the real final clusters much faster than the original procedure, while keeping (and even improving) the memory and speed qualities of the original procedure. Based on previous work’s results, we have decided to fix the Hoeffding bound confidence level parameter to \( \delta = 0.05 \). We will consider a different confidence level for the assignment criterion, with parameter \( \delta_a \).

5.1 Artificial data

We designed a three clustered data set, where all the clusters are equidistant, which was possible by creating three streams completely uncorrelated and adding noise to two copies per original variable, therefore becoming highly correlated with the original. Figure 2 shows the resulting structures, using strict (left) and confident (right) assignment. As feared, the original procedure made a first-level splitting where one of the clusters was erroneously split, resulting in a final structure with four clusters. Our procedure found the three clusters fast and accurately, supporting our arguments. But what if the clustering structure is more complex? We designed a different data set, with four clusters again uncorrelated. Figure 3 shows the resulting structures, using strict (left) and confident (right) assignment. The original technique revealed some instability due to erroneous splits (which forced later aggregations) before ending with the correct result. The confident assignment produced a correct answer quicker, with no aggregations. Moreover, the outcome hierarchy is less deep and much more readable, as we can quickly acknowledge which clusters forced each split. Also, in this data set we can observe that the system only made a threefold split in the first level, generating three siblings, while in the second level the split was binary as the bounds were fully observed for all the remaining variables. From this, we can stress that our approach will decide at each level if the data is symmetrically separable or not.

5.2 Real data

In order to assess the improvement given by this new assignment strategy, we have applied the algorithm to real world data. Time series of electrical data are one of the most widely studied sets of data. From the raw data received at each sub-station, taking into account only the current intensity sensors, we aggregated observations on a hourly basis over more than two and a half years. This data set represents 2700 sensors along 22364 observations.

The proposed method relies on a confidence test based on the Hoeffding bounds. This way, sensitivity analysis must be performed to assess the level of dependence of the method to this parameter. Figure 4 presents the analysis on a small sample of the current intensity data set, with 51 variables observed along two years. We can observe that, for chosen values of \( \delta_a \), the system reveals low sensitivity, being the best results observed for low \( \delta_a \) parameter values, which means...
that only variables which are closer to one of the pivots with high confidence should be assigned. The experiment with $\delta_a = 1$ applies the original symmetric assignment criterion. Although highest values of DVI were reported for $\delta_a$ values of 0.02 and 0.03, the best hierarchy (given by the CPC value) occurred for $\delta_a = 0.01$. Nevertheless, we believe that considering low levels of $\delta_a$ will always be preferred to the symmetric assignment criterion, as this resulted on poor validity indices. From this point, we chose to fix $\delta_a = 0.01$.

For the complete current intensity data set we ran two experiments, using the original symmetric assignment criterion and the new confident assignment criterion. We monitored both the update speed and memory usage along time, as these are two of the main positive arguments of the proposed strategy. The objective of this experiment was to assess the increase in performance of the entire system. In Figure 5 we present tendency graphs based on weekly averages of the performance measures: stored values, for memory usage monitoring; and examples per second, for speed monitoring. We can observe how the overall processing speed and memory usage improve with time, especially for the proposed strategy. It becomes clear the advantages of threefold assignment in the speed-up achieved by splitting. Moreover, even with the increasing number of clusters to test, the total processing speed increases with splits, reducing the total amount of memory needed to operate.

6 CONCLUDING REMARKS

In this paper we have presented a confident assignment criterion for divisive clustering of streaming time series. In this task series, the goal is to find clusters of similar time series over time. Previous works have already addressed the problem of assigning confidence to the decision of the moment of split. The split decision used in some divisive systems focuses on the symmetrical boundary between two splitter pivot variables, generating uncertainty in the assignment since it has to decide based on only a small subset of the entire data. Our proposal is to include a confidence level to the consequent assignment process. When a split point is reported, creating two new clusters, we can directly assign points which are confidently closer to one cluster than the other, having a different strategy for those variables which do not satisfy the confidence level. In this work we have proposed a robust approach to the assignment of variables to newly created clusters, enabling the creation of a third sibling with all variables for which confidence bounds are not verified. Experimental evaluation is presented in the context of the application of this criterion in ODAC, a recently proposed hierarchical algorithm.

However, this approach can easily be applied to any divisive procedure other than ODAC. Experimental results show that this new assignment criterion will find better hierarchies quicker, with the outcome tree being also much more readable. Our approach will decide at each splitting level if the data is symmetrically separable or not, creating two or three new clusters accordingly. The main advantages of this criterion include a better decision on the assignment of unsure variables, while keeping memory usage and processing time requirements. Current and future work is concentrated on extending experiments to different datasets and systems, and the clarification of the impact of this new setup on structural drift detection feature.

ACKNOWLEDGEMENTS

The work of Pedro P. Rodrigues is supported by the Portuguese Foundation for Science and Technology (FCT) under the PhD Grant SFRH/BD/29219/2006. The authors thank the Pluriannual financial support attributed to LIAAD and acknowledge the participation of FCT Project ALES II under Contract POSC/EIA/55340/2004. We would also like to thank the referees for their comments which helped improve this paper.

References