

Similarity Measurement in Recent Biased Time Series Databases using Different Clustering Methods

D. Muruga Radha Devi^{1*} and P. Thambidurai²

¹Research Scholar, Sathyabama University, Chennai–119, India; emrdevi@hotmail.com

²Department of CSE, PKIET, Karaikal, India; ptdurai58@yahoo.com

Abstract

Time series data are commonly used in data mining. Clustering is the most frequently used method for exploratory data analysis. In this paper a model is proposed for similarity search in recent biased time series databases based on different clustering methods. In recent biased analysis, data are much more interesting and useful for predicting future data than old ones. So in our method, we try to reduce data dimensionality by keeping more detail on recent data than older data. Due to “Dimensionality Curse” the original data is mapped into a feature space by means of Vari-segmented Discrete Wavelet Transform¹ and then similarity measurement is performed by applying different clustering methods like Self Organizing Map (SOM), Hierarchical and K-means Clustering. This model is tested using Control Chart Data and the clustering result observed proves that the proposed model is better in grouping similar series under various resolutions.

Keywords: Clustering, Dimensionality Reduction, Discrete Wavelet Transform, Feature Extraction, Hierarchical Clustering, K-means Clustering, Self Organizing Map, Similarity Measurement

1. Introduction

The increasing use of time series data has initiated a great deal of research in the field of data mining. Various kinds of time series data related research are for example, finding similar time series, subsequence matching, dimensionality reduction and segmentation. Time series data is naturally large in size, high dimension and has to be updated continuously. Therefore, unlike traditional databases where search is for exact matching, in time series data it is carried out in an approximate manner. In time series data mining, the fundamental problem is in its proper representation. One of the common methods is transforming the time series to a reduced domain by dimension reduction and measuring similarity between time series or subsequent series for different mining tasks. To measure the similarity between two time series, the most popular approach is to measure the

Euclidean distance on the transformed representation like the DFT coefficients and the DWT coefficients².

The problem of clustering in the time series domain finds applications like grouping entities with similar trends. The determination of clusters of time series is extremely challenging because of the difficulty in determining similarity among different time series, which are scaled or translated differently on various dimensions. Therefore, the concept of similarity is a very important one for time-series data clustering. Accordingly, an appropriate clustering algorithm and distance measure should be chosen. For example, Euclidean distance reflects similarity in time, while Dynamic Time Warping (DTW) reflects similarity in shape. A significant difference in clustering between time-series data and of clustering objects in Euclidean space is that the time series to be clustered may not be of equal length.

*Author for correspondence

Time series clustering has applications in different domains namely:

1. In financial markets, the values of the stocks represent time series which change with time and by clustering such time-series details insights into the data can be obtained.
2. Different kinds of medical data which when clustered provide an understanding of the data which can be related to different kinds of diseases.
3. Numerous applications in earth science, such as temperature or pressure or water level recording in lakes determine the frequent trends in the data which can provide idea about the common climatic condition.

1.1 Our Contributions

In this paper, we have extended the similarity measurement model for recent biased time series databases¹ which we have designed initially only with SOM Clustering and also by applying K-means Clustering and Hierarchical Clustering. Pre-processing is performed on all the series before applying clustering, since this reduces the complexity involved in distance calculation for the similarity measure. This model has been tested using control chart time series for which the various categories are already known. Similar type of clustering is observed on the simulated series that has been considered for the verification.

2. Background and Related Work

First step in a clustering analysis task is to define similarity together with feature selection. The similarity between two series in the feature space can be determined by two parameters: Distance and Similarity Measure.

2.1 Distance

The similarity of two series can be measured by distance between them. There are a number of such distances, which could be used to measure the similarity of the series. Among the various distance metrics, Euclidean Distance is the one that is most widely adopted in practice. We can select different distance measures, depending on the kind of data used in clustering³.

Minkowski Distance is the generalization of several well-known distances which is given by

$$D_{ij} = \left[\sum_{l=1}^d |x_{il} - x_{jl}|^{1/n} \right]^n \tag{1}$$

(i.e.,) if $n = 1$, it is the 'City block' distance; if $n = \infty$, it is the 'Chebychev' distance and when $n = 2$, it is the Euclidean Distance (i.e.)

$$D_{ij} = \left[\sum_{l=1}^d |x_{il} - x_{jl}|^{1/2} \right]^2 \tag{2}$$

2.2 Similarity Measure

Similarity measure is of fundamental importance for time series analysis and data mining tasks. Most of the methods propose the similarity measure on the transformed representation scheme. In traditional databases, similarity is based on exact match between the data, but in time series data, similarity measure is carried out in an approximate manner. The time-series clustering task can be divided into two categories and the query results are expected to provide useful information for different analysis activities⁴.

Whole Sequence Clustering: Clustering can be applied to each complete time series in a set.

Subsequence Clustering: Clusters are created by extracting subsequences from a single or multiple longer time series.

For e.g. consider the stock time series facing queries like:

- Query 1:** find all stocks which are "similar" to stock A.
- Query 2:** find all patterns last for a month in the closing prices of all stocks.

With respect to Query 1 and Query 2 above, they can be considered as a whole sequence matching and a subsequence matching, respectively. Gavrilov et al⁵. has presented the usefulness of different similarity measures for clustering similar stock time series.

2.2.1 Similarity Measuring Criteria

Similarity between two clustering methods is measured using the following formula:

Given two clustering, one as benchmark $C = C_1, C_2, \dots, C_k$ and other as clustering method that we are applying i.e., K-means or Hierarchical or SOM Clustering, $C' = C_1, C_2, \dots, C_k$ ⁵

$$Sim(C_i, C_j) = \left(\sum \max(Sim(C_i, C_j')) / k \right)$$

$$Sim(C_i, C_j') = \frac{2|C_i \cap C_j'|}{|C_i| + |C_j'|} \tag{3}$$

This similarity measure will return 0 if the clusters are completely different and returns 1 if they are same.

2.3 Feature Extraction Methods

Feature extraction is used for holding unique features and avoiding redundancies. So if the right features are extracted, time series will be reduced to selected features, that represents part or the whole series and data mining algorithms will be executed fast and also yields better results than using original data¹.

The work by Agrawal et al.⁶, establishes the representation of time series as a set of coefficients obtained from a Discrete Fourier Transform (DFT) to reduce the dimensionality of data. This paper laid the foundation for many subsequent works which were enlarged by using properties of the DFT or similar decompositions with similar efficiency such as Discrete Wavelet Transform (DWT)². Keogh and Faloutsos et al.⁷ proposed Piecewise Aggregate Approximation (PAA) which suggested approximating a time series by dividing it into equal length segments and by recording mean value of the data points that fall within the segment as a sophisticated transform. Keogh et al.⁸ also introduced Adaptive Piecewise Constant Approximation (APCA) wherein the segments have arbitrary lengths and two numbers per segment, the first records the mean value of the segmental data points, while the second records the length.

We have already designed the similarity measurement model by applying SOM clustering alone and tested the model using stock series and in that work for feature extraction Vari-segmented DWT method is used. In this method the time series is divided into varying length segments and DWT is applied on all the segments to extract equal number of coefficients from each segment so that more number of coefficients retained for recent segments and less number of coefficients for old segments which would be helpful for recent-biased analysis¹.

2.4 Clustering Time Series

Clustering is combining points by the concept of 'closeness' or 'similarity' in various ways, according to the previous knowledge of the problem. Cluster analysis aims to group data items into clusters, wherein items within a cluster are more 'similar' to each other than to the items in the other clusters. Cluster analysis is widely used in varied applications like data mining, statistical data analysis, information retrieval, pattern recognition, image processing, and bioinformatics.

Clustering is traditionally an unsupervised learning process since it is performed when no information is

available concerning the membership of data items. A single partition of the collection of items into clusters is referred as Partitional Clustering, whereas obtaining a hierarchy of clusters is referred as Hierarchical Clustering. Some methods rely on representations of the data to define prototypes and data distributions besides computing similarities. Other methods only require the evaluation of pair wise similarities between data items; while imposing less restrictions on the data; these methods usually have a higher computational complexity. A classification of clustering methods is proposed in Han and Kamber⁹ showing five categories: Partitioning, Hierarchical, Density based, Grid based, and Model based.

2.4.1 K-means Clustering

K-means is a divisive, non-hierarchical and partitional method of defining clusters. This is a repetitive process, wherein at each step the membership of individual in a cluster is re-evaluated based on the current centers of each existing cluster. This is repeated until the desired number of clusters is reached. Thus, it is non-hierarchical because an individual can be assigned to a cluster, and reassigned to others at any later stage in the analysis. The algorithm converges when the assignments no longer change.

The K-means algorithm applies to objects that are represented by points in a d-dimensional vector space into k clusters of points. That is, the k-means algorithm clusters all of the data points in D such that each point X_i falls in one and only one of the k partitions¹⁰, i. e. given a set of points, the single best representative for this set is the one that minimizes the Sum of the Squared Euclidean (SSE) distances between each point and the mean of the data points. The number of iterations required for convergence varies and may depend on N where each iteration needs $N \times k$ comparisons.

The algorithm is sensitive to the initialization method and can lead to local minimum. Choosing the optimal value of k may be difficult, but with the knowledge of the dataset, such as the number of partitions that comprise the dataset, then that can be used to choose k. K-means is order independent, (i.e) for a given set of cluster centers, it generates the same partition of the data irrespective of the order in which the patterns are presented to the algorithms¹¹. Time complexity of K-means clustering is $O(nkl)$ where 'n' is the number of patterns, 'k'

the number of clusters and 'l' is the number of iteration taken by the algorithm to converge and Space complexity is $O(k+n)$ and also an additional space for storing data matrix.

2.4.2 Hierarchical Clustering

Hierarchical clustering is used to group similar objects into 'clusters' where each row or column is considered a cluster. Hierarchical clustering returns a sequence of nested partitions, where each increasing level merges two cells of the lower level, showing a clustering hierarchy which enables us to predict how close two clusters are present³.

Hierarchical Clustering is divided into two categories namely:

- i. Agglomerative methods, which proceed by series of merging of the objects into groups and this is also named as "bottom up", since small clusters are grouped into larger ones.
- ii. Divisive methods, which separate objects successively into finer groupings and are also named as "top down", since it splits big clusters into small ones.

The two most widely used distance measures in hierarchical clustering are:

- Single linkage clustering (nearest neighbor): distance between groups is defined as the distance between the closest pair of objects, where only pairs consisting of one object from each group is considered i.e. the distance between two clusters is given by the value of the shortest link between clusters. At each stage the two clusters for which the distance is considered less are merged.
- Complete linkage clustering (farthest neighbor): is the opposite of the single linkage i.e. distance between groups is defined as the distance between the most distant pair of objects, one from each group.

Advantages of Hierarchical Clustering are its flexibility in handling any form of similarity or distance and algorithm is more versatile. Major weakness of Agglomerative Clustering methods is that they do not scale well and time complexity is at least $O(n^2)$, where n is the number of total objects and they can never undo what was done previously¹¹.

2.4.3 Self Organizing Map

Kohonen in 1981 proposed Self Organizing Map (SOM), an unsupervised learning algorithm. SOM is both a

projection and a clustering method, in which similar data samples are mapped to nearby neurons^{12,13}. SOM consists of 2-D grid of map units which are connected to adjacent ones by a neighboring relation. Map units vary from few dozen to several thousand, indicating the generalization capability of SOM. In SOM, data points lying near each other are mapped onto nearby map units and referred as a topology preservation mapping. The important property of SOM is that it forms a non-linear projection of high dimensional data into a low dimensional 2-D grid. Two levels clustering by SOM proposed by Juha Vesanto et al.¹⁴ suggests that clustering SOMs instead of clustering data is computationally effective.

2.4.4 Data Output Visualization

Initial idea of number of clusters in SOM and their spatial relationship is identified by visual inspection of the map. Unified Distance Matrix Techniques (U-Matrix) is a widely used method for visualizing cluster structure of SOM, showing distances between prototype vectors of neighboring map unit by using gray scale¹⁵. Light color indicates smaller distance between neighbors, while dark color indicates larger distance. SOM training, positions these interpolating map units between clusters as borders. The quality of clustering depends not only on the similarity measure but also on the clustering algorithm used. Another method to display number of clusters is the SOM-hits in each map unit. Interpolating map units have very few SOM hits or may even have zero hits indicating cluster borders. Advantages of SOM clustering are that different kinds of distance measures and joining criteria can be used to form big clusters¹.

3. Similarity Measurement Model

In stream data analysis, users pay more attention to recent data, and are often interested in recent changes, rather than long-term changes. So it is reasonable to process time series data with emphasis on recent values since space requirement will be much reduced and the querying on time series will be more efficient, which is referred as Recent-biased Analysis.

At first the input time series is segmented according to the resolution levels (i.e.) in increasing powers of two and then feature extraction is applied uniformly on all segments. If equal number of coefficients is selected from all segments, the recent segments whose size is small, more information will be maintained and old segments where

the segments size is large less information will be stored. Then the extracted features that are considered as best representatives of the time series considered are given as input to clustering for similarity measurement process.

For high dimension datasets, cluster exists in some subspaces and moreover distance measure also becomes meaningless since all vectors are equidistant to the search query vector. So dimension reduction is performed as a pre-processing step. If representative features are determined using dimension reduction, then the cluster formation will be clear. Feature extraction is used for holding only best features and getting rid of redundancies. Similarity measurement model is designed using three types of clustering namely K-means, Hierarchical and SOM.

Algorithm for Similarity Measurement Model

Input: Raw Time series $S_1, S_2 \dots S_N$

Output: Result of different clustering methods applied

- i) Feature extraction using Vari-segmented DWT.
- ii) Clustering Method Selection
 - a) K-means Clustering
 - b) Hierarchical Agglomerative Clustering (HAC)
 - c) Self Organizing Map (SOM)
- iii) Compare the performance of similarity mapping returned by the three methods on original and reduced data.
- iv) If the performance is accepted then Return Clustering Result else Return to clustering method.
- v) Repeat the process with the simulated series and verify the clustering results.

3.1 Feature Extraction using Vari-segmented DWT

Clustering is a common method for finding similarity in given data. Clustering algorithms depend on meaningful distance function to group data vectors that are close to each other. But in high dimensional spaces it is very difficult to find meaningful groups. So each time series is transformed into reduced domain and best coefficients are used in clustering for determining similarity. A reduction of time series into few features also increase the analytical value of the results and the clustered results show the mutual dependencies between the variables and dataset.

Steps in the feature extraction process using Vari-segmented DWT are:

- i) Time series is divided into segments, where recent data are partitioned into smaller segments to keep more details and larger segments can be set for older data, so that less detail is kept for them. The size of the segments is set in powers of two because it is more space efficient and DWT run fastest with this length. Therefore the length of segment S is set to $n_i = 2^i$ for $i = 1, 2, 3 \dots n$ or it can be set to any increasing number sequences.
- ii) After partitioning the series, DWT is applied to each segment and the same number of coefficients is selected from every segment.
- iii) Best coefficients from each segment is considered as the representatives of the series and is taken as input for the clustering methods¹.

3.2 Clustering Methods

Clustering of time series data, like clustering for all types of data, has the goal of producing clusters with high similarity between objects within cluster and low similarity between different cluster objects. In time series clustering it is crucial to decide what kind of similarity is important for the clustering application.

3.2.1 K-means Clustering

The K-means algorithm gives us a partition, because it just gives us a single set of clusters, with no particular organization or structure within them. Initial number of groups or clusters need to be specified. Since initial cluster assignment is random different runs of K-means clustering algorithm may not end up with the same final solution. To solve this, K-means algorithm is repeated many times where each time starts with different initial clusters. The sums of distances within the clusters are used to evaluate different clustering solutions. The solution with smaller sum of within-cluster distance is considered as optimal solution. If optimal solution is found more than one time, then the algorithm has found overall optimal solution where SSE value is minimum.

3.2.2 Hierarchical Clustering

Basic idea in Hierarchical Clustering is to arrange set of items into a tree called *Dendrogram*, where items that are joined by short branches are very similar to each other and by increasingly longer branches for decreased similarity. Given a set of N items for clustering along with $N \times N$ distance matrix, the Agglomerative Hierarchical

Clustering process undergoes repeated cycles where two closest items are joined by a branch of a tree, with length of the branch set to the distance between the joined items. The two joined items are removed from list of items being processed and replaced by an item that represents the new branch. The distance between new item and all other remaining items are computed, and the process repeated until only one item remains. A partition of the data from a hierarchy can be obtained by cutting the tree of clusters at some level.

3.2.3 SOM Clustering

SOM is an unsupervised network where the target value need not be specified. In this network (Figure 1) where the node weights match the input given, we can say that area of the network closely match the input value. Initially random weights are assigned to the nodes of the network, after much iteration the SOM settles into a map of stable zones, where each zone acts as a feature classifier. If the initial weights are not chosen properly SOM generates sub-optimal partition. The objective is not to find optimal solution but get good insight into the cluster structure of the data for data mining purposes. Due to generalization property, previously unseen input vectors presented to the network for testing purpose will activate similar weight vectors nodes, namely its neighbors.

SOM performs very well in forming clusters even with distance greater than HAC method. SOM applies to large datasets but the computational complexity grows with number of data samples. It does not require large amount of memory but training takes more time which can be speeded up by implementing improvised algorithms.

4. Experimental Results

4.1 Data Set

The performance of the similarity model is tested using the Control Chart Dataset¹⁶, which consists of 600 series

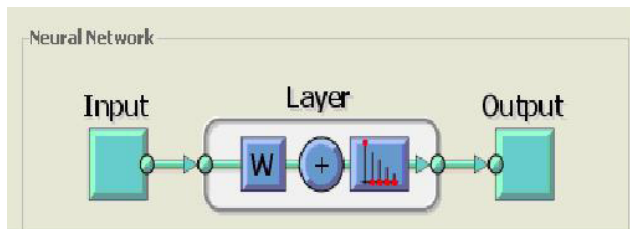


Figure 1. Network structure of SOM clustering.

where each time series consists of 60 data points. First, the model is trained with 24 series selected from different categories and then the same similarity process is tested with the rest of the series. Since this is the benchmark data the number of clusters is already known. So we have considered this data set for testing our model. The data of these series falls into six categories namely: Normal, Cyclic, Upward Trend, Downward Trend, Upward Shift and Downward Shift.

Each series is divided into segments of increasing powers of two and then DWT applied on each segment and then the representative DWT coefficients from each segment is taken as best feature for the series representation. So the reduced series consists of best coefficients from each series. The reduced form of this sample data is the feature space which forms input to the clustering process.

This model has implemented clustering methods by using the facility provided by MATLAB, for e.g., using K-means procedure for K-means clustering and for Hierarchical Clustering by invoking Dendrogram plotting functions with necessary arguments and “nc tool” facility for SOM clustering. Performance of the three clustering methods were compared and analyzed by using the following supporting figures which we obtained by executing the procedures using the dataset mentioned above.

4.1.1 K-means Clustering

Original data was applied to K-means Clustering (Figure 2) directly since it works fine with huge dataset. This also gave an initial idea about number of clusters or categories in the entire dataset. But the boundary between clusters were not that much clear and so reduced data was given as input to the clustering again where cluster formation was very clear. The plot with the reduced series (Figure 3) verifies the similarity with the series and efficiency of the reduction process.

4.1.2 Hierarchical Clustering

Dendrogram shows the clustering hierarchy only for about 30 series at a time. So from the original data, 4 series from each category were considered for HAC as sample data (24 × 60) (Figure 4). Since data computation is more for calculating the distance matrix, it is a time consuming process and the clustering result is also very ambiguous. For the reduced series closest pairs are easily identified from the tree shown and then it is also tested

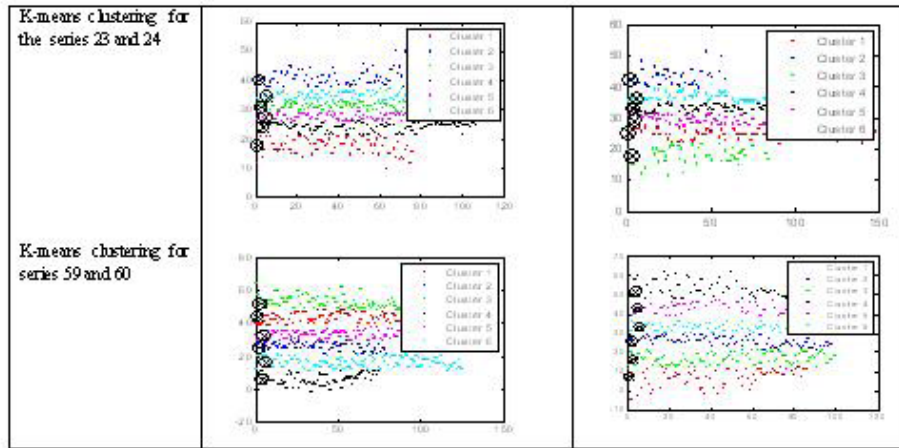


Figure 2. K-means clustering with the original data series.

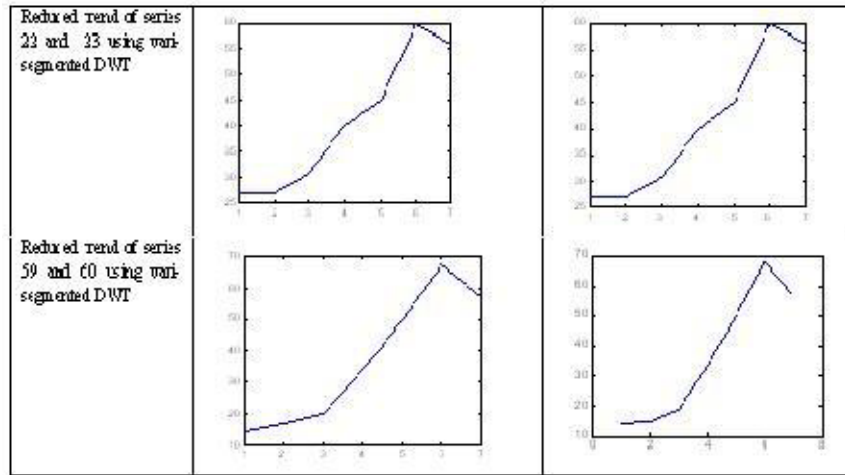


Figure 3. Similarity trend for the reduced series.

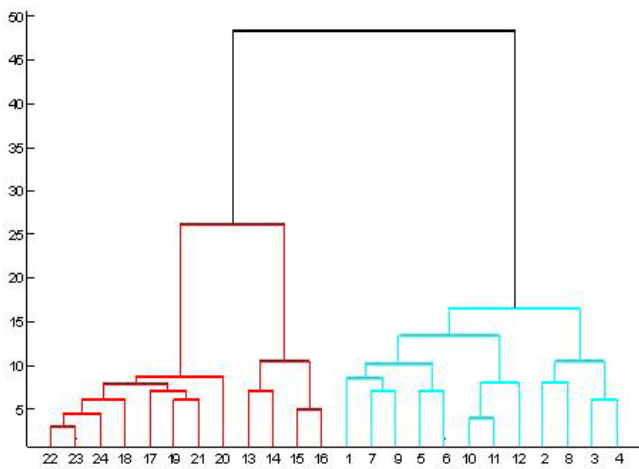


Figure 4. HAC result for the sample data (24 series).

with the similarity trend drawn with the series. Next we applied simulated data consisting of 10 series (Figure 5) from various categories and then with 20 series (Figure 6) which includes the previous 10 series also for testing purpose and the clustering formed are also observed.

The result obtained from HAC was taken as a hint to fine tune the SOM process to obtain optimal clustering during the training process. i.e., the sample series that is taken for clustering is first applied to HAC then by observing the number of groups formed and based on the levels, we verified the clustering output with SOM.

4.1.3 SOM Clustering

Initially from the original data, 4 series from all the six categories i.e., 24 series are considered for clustering

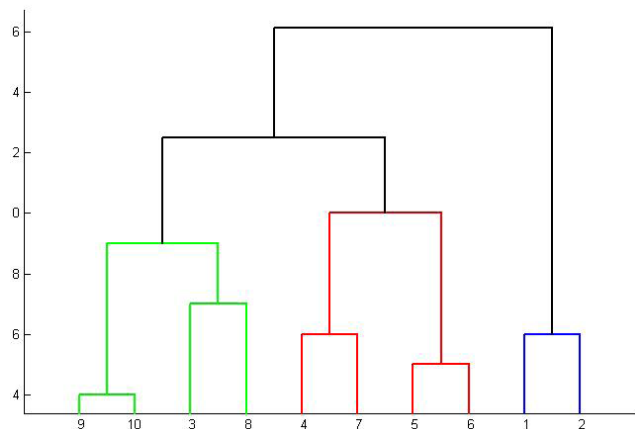


Figure 5. HAC result for the simulated data (10 series).

without applying pre-processing methods like reduction. Here convergence took more time since identifying neighborhood and determining clustering is time consuming for more volume of data. So, pre-processed data that is having best coefficients from each segment is considered for the clustering process.

First we started SOM method with 6×6 nodes for the sample data. Since more nodes are having zero hits, optimum clustering was not observed. So we tried reducing the No. of nodes from 6×6 to 5×5 to 4×4 (Figure 7) and then to 3×3 (Figure 8) where most of the nodes are filled with hit details and also optimal clustering was observed. Same process repeated with 20 series i.e., (6×6 to 5×5 to 4×4 (Figure 9) and then to 3×3 (Figure 10)) which also includes previous 10 series. Here also only with 3×3 nodes optimal clustering was observed.

We have selected three clustering methods for this model namely K-means Clustering, Hierarchical Agglomerative Clustering (HAC) and SOM due to their popularity, flexibility, applicability and handling high dimensionality and our experiments with the Control Chart Data and simulated data also verifies the following facts which are already studied by using clustering software in¹¹.

- As no. of clusters increases performance of SOM decreases whereas K-means is better than Hierarchical clustering in this case.
- K-means algorithm finds a clustering solution with a lesser distance than the hierarchical clustering techniques.
- SOM shows more accuracy in classifying the objects to their clusters if k is small but if k increases HAC

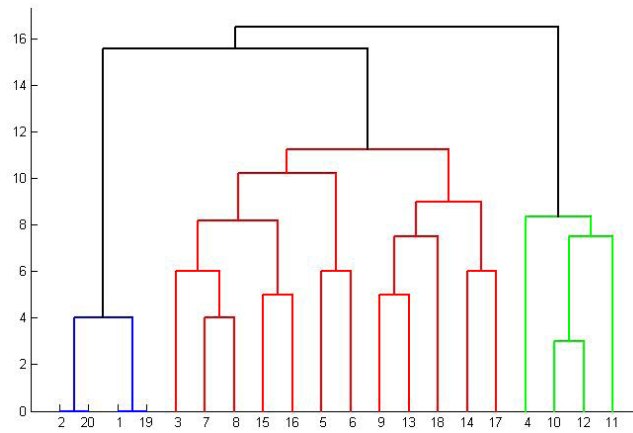


Figure 6. HAC result for the simulated data (20 series).

becomes better and K-means is less accurate than other two methods if k increases.

- K-means shows good performance for huge dataset whereas SOM and hierarchical clustering shows good result for small dataset since computation of distance matrix is time consuming for HAC and convergence takes lot of time in the case of SOM. So these methods work well on the reduced data very well.

5. Conclusion

A similarity measurement model has been developed for recent biased time series databases by applying Vari-segmented DWT to reduce the dimension, then applying different types of clustering like K-means, Hierarchical and SOM. We have tested this model using control chart time series. The clustering result shown by the hierarchical clustering method is considered as reference to compare the performance with the SOM method. K-means clustering works with huge data set but experiments prove that identification of clusters is very difficult by using original data directly. Moreover distance computation with original series is very tedious and then visualizing the clusters with raw series for high dimension is a restriction in both the cases of SOM and HAC. So in this paper the clustering performance of the model proposed over reduced series using feature extraction is observed and tested with the control chart data set. The simulated result proves that the similarity measurement with SOM clustering is better in grouping similar series under various resolutions than the K-means and hierarchical methods.

K-means works well in large data set and Hierarchical clustering is simple but computation intensive method

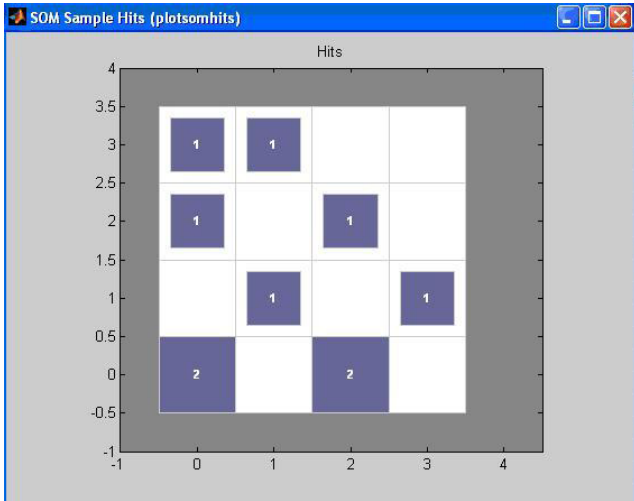


Figure 7. Sample hits for SOM clustering using 4×4 grid (10 series).

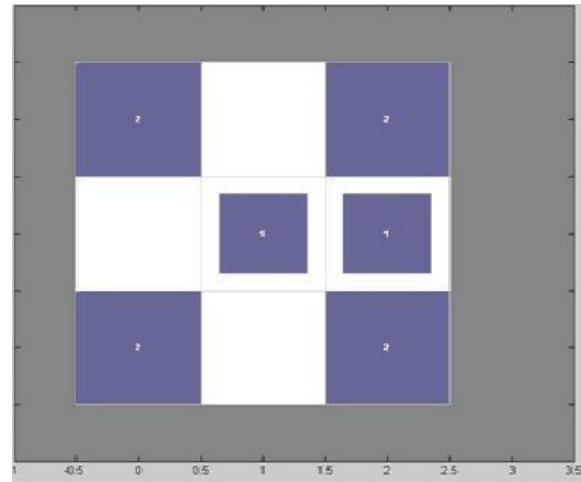


Figure 8. Sample hits for SOM clustering using 3×3 grid (10 series)

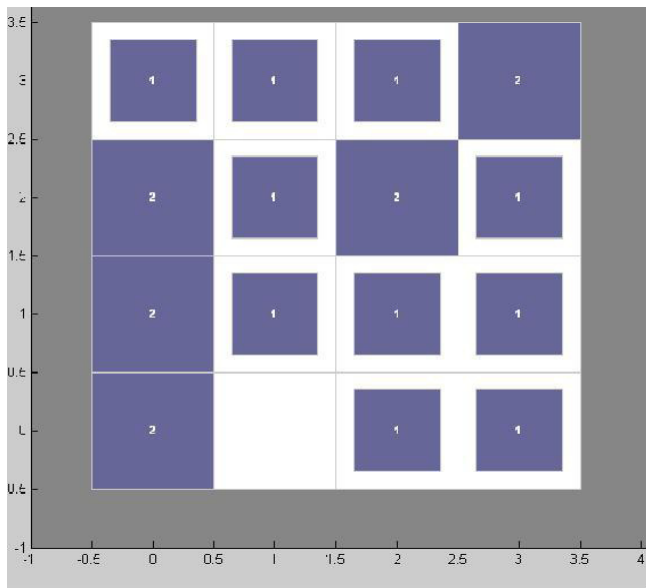


Figure 9. Sample hits for SOM clustering using 4×4 grid (20 series).

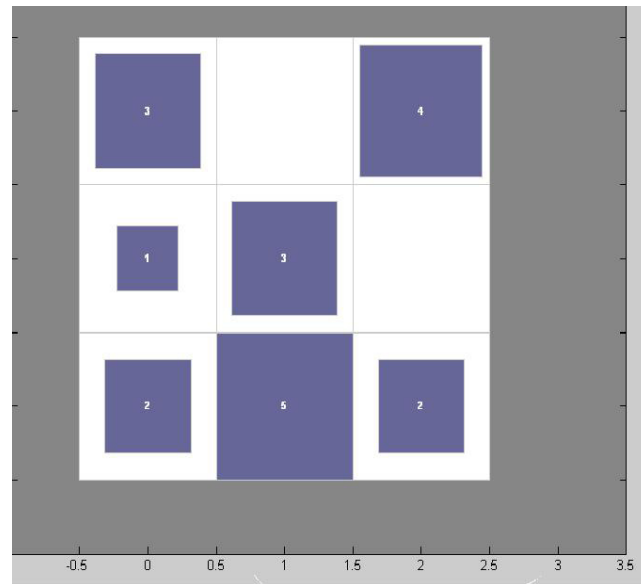


Figure 10. Sample hits for SOM clustering using 3×3 grid (20 series).

so used for verification purpose but cannot be scaled for large data.

6. References

1. Radha Devi DM, Thambidurai P. Clustering based similarity measurement model for recent biased time series databases. *International Journal of Computer Science Engineering and Information Technology Research (IJCEITR)*. 2013 Aug; 3(2):355–62.
2. Morchen F. Time series feature extraction for data mining using DWT and DFT. Philipps–University Marburg; 2003. Technical Report No. 33
3. Rokach L, Maimon O. *Data Mining and Knowledge Discovery Handbook*. Part III. US: Springer; 2005. Chapter 15, Clustering Methods; p. 321–352.
4. Fu TC. A review on time series data mining. *Eng Appl Artif Intel*. 2010; 24(1):164–81.
5. Gavrilov M, Anguelov D, Indyk P, Motwani R. Mining the stock market: which measure is best? *Proceedings of*

- the Sixth ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2000; 487–96.
6. Agrawal R, Faloutsos C, Swami A. Efficient similarity search in sequence databases. In: Lomet DB, editor. Proceedings of the 4th International Conference on Foundations of Data organization and Algorithms; 1993 Oct 13–15; Chicago, Illinios, USA: Springer Verlag; 1993. p. 69–84.
 7. Keogh E, Chakrabarti K, Pazzani M, Mehrotra S. Dimensionality reduction for fast similarity search in large time series databases. *Knowl Inform Syst*. 2001 Aug; 3(3):263–86.
 8. Keogh E, Chakrabarti K, Pazzani M, Mehrotra S. Locally adaptive dimensionality reduction for indexing large time series databases. *ACM Transactions on Database Systems*. 2002 Jun; 27(2):188–228.
 9. Han J, Kamber M. *Data mining concepts and techniques*. 2nd ed. Morgan Kaufmann Publishers. 2009.
 10. Xu R, Wunsch DT. Survey of clustering algorithms. *IEEE Trans Neural Netw*. 2005 May; 16(3):645–78.
 11. Abbas OA. Comparison between data clustering algorithms. *The Int Arab J Inform Tech*. 2008 Jul; 5(3):320–25.
 12. Kohonen T. *Self-organizing maps*. 2nd ed. Berlin, Heilderberg, Germany: Springer; 1995.
 13. Zhang P, Li X, Zhang Z. Similarity search in time series databases based on SOFM neural network. *Third International Conference on Natural Computation, ICNC 2007*. 2007 Aug 24–27; Haikou, China; 2007. p. 715–18.
 14. Vesanto J, Alhoniemi E. Clustering of SOM. *IEEE Trans Neural Netw*. 2000 May; 11(3):586–600.
 15. Vesanto J. SOM based visualization methods. *Intell Data Anal*. 1999; 3(2):111–26.
 16. UCI KDD Archive. Available from: <http://kdd.ics.uci.edu/>