Classification of Data Sequences by Similarity Analysis of Recurrence Plot Patterns

E. Bautista-Thompson  
R. Brito-Guevara  
Centro de Tecnologías de la Información  
DES-DACI, Universidad Autónoma del Carmen  
Ciudad del Carmen, Campeche, México  
ebautista@pampano.unacar.mx  
rbritoguevara@hotmail.com

Abstract

Quantification of the similarity between data sequences is important in different database and data mining tasks such as indexing, retrieving, clustering and classification, many similarity metrics (Euclidean, DTW, among others) operate directly in the raw representation of the sequences, but this implies when the sequences are compared, not take into account information about the collective behavior of the data that forms a sequence and the hidden relations between such data, such information can be important for classification of sequences based on their structures and their relations with the dynamics that such structures can represent (e.g. stationary, random, complex). We propose a computational technique for similarity analysis and classification of recurrence plot patterns: RecurrenceVs. The results show that the proposed technique is able to classify data sequences by similarity families based on the recurrence plot patterns, which preserve the information about the structure and dynamics represented by the data sequences.

1. Introduction

The extraction of common structural features in sets of time series and sequences of data is an important task in the identification of patterns of interest for example: in the analysis of time series dynamics [1], the identification of motifs in genomic sequences [2], the construction of queries for sequence extraction in databases [3]. Similarity in data sequences can be measure with different metrics such as Lp Norms [4], Dynamic Time Warping [5], Similarity Histograms [6], etc.; such measures operate directly over the data sequences. For other side, analysis of the dynamics of sequences of data, understood as the relationships between the different data inside a sequence, can be done with different techniques for example: Singular Spectral Analysis [7], Empirical Mode Decomposition [8], and Recurrence Quantification Analysis [9]. In particular, the recurrence quantification analysis is based on the generation of a data representation named recurrence plots, these plots shows hidden relationships between data such as non-stationary behavior, periodicities, and randomness; the patterns generated with these plots can be used to compare sequences of data, at a new level of information. There are not quantitative comparison tools for recurrence plots, software such as VRA, RQA and CRP lacks this functionality [10, 11, 12], this is the main motivation for the development and proposal in this work of a computational technique for quantitative comparison and analysis of similarity between recurrence plots. Section two, explains the theoretical basis of the recurrence plots. Section three, describes the algorithmic aspects of RecurrenceVs, the software tool that implements the technique. In Section fourth, the experimental results of the evaluation of the software tool are presented. Finally in section five, the conclusions of this work are presented.

2. Recurrence plots

Recurrence plots are graphical representations of sequences of data, which allows the detection of hidden dynamical patterns and nonlinearities inside the data. These plots allow the visualization of recurrent patterns, non-stationary patterns, and structural changes. The recurrence plots are part of a technique known as Recurrence Analysis that was developed by Eckman et. al. in 1987 [9]. The patterns generated by the recurrence plots multiplies the information of a sequence of data (time series, genomic sequence, etc.), a fundamental idea behind a sequence of data that
represents a measure of an observable from a particular system is that this set of data contains multidimensional information about such system [13, 14].

A recurrence plot is generated by comparison of each datum in a sequence with itself and with the rest of data.

\[ x(i) = \{x(1), x(2), x(3), \ldots, x(n)\} \] (1)

This comparison is made with a metric such as Euclidean distance; this allows to build a correlation matrix of spatial and temporal nature (as is the case of time series).

\[ d_{ij} = \|x(i) - x(j)\| \] (2)

\[ D = \begin{bmatrix}
    d_{11} & d_{12} & \cdots & d_{1n} \\
    d_{21} & d_{22} & \cdots & d_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    d_{n1} & d_{n2} & \cdots & d_{nn}
\end{bmatrix} \] (3)

Each element \( d_{ij} \) in matrix \( D \) is associated with the Euclidean distance between a datum in position \( i \) and a datum in position \( j \) inside a sequence, if a datum or subsets of data are recurrent this behavior will be detected by means of sets of equal distances. Also, each distance \( d_{ij} \) is associated with a value from a discretized gray tone scale, in this way an image of the recurrence plot pattern is generated. The resultant pattern is symmetric due to the redundancy in the calculation of the distances \( d_{ij} = d_{ji} \).

Comparing the patterns generated by the recurrence plots can be useful in order to identify sequences of data with similar behavior or dynamics, this similarity must be quantified in order to have an objective comparison of the recurrence plots and be able to classify them. In the next section the algorithms for the comparison method are described.

3. RecurrenceVs: algorithms

The design of the algorithms was divided in three stages: first the load and preparation of the data, second the construction and comparison of the recurrence plots and third the visualization and storage of the results. In the first stage, an important aspect if the normalization of the data sequences to be compared, the data sequences can have different length, and data range (the data types used are: integer and real), in order to generate comparable recurrence plots they are normalized to a default length of 1000 data by sampling each sequence, but this parameter of length can be adjusted or not used if shorter time series are used in the similarity analysis. The normalization in range is done by scaling the values in order to have a range of values between 0 to 1; the Figure 1 shows a diagram of this stage.

The construction of the recurrence plots is made as was described in section two. The comparison between two recurrence plots is made by comparing the corresponding vector elements \((i, j, d_{ij})\) of the matrix of distances \( D \) for each one of the recurrence plots, for each position the difference between the distances \( d_{ij} \) is calculated.

\[ \text{diff}_{d_{ij}} = \|d_{ij_{rp1}} - d_{ij_{rp2}}\| \] (4)

A distance \( d_{ij} \) is considered as a match between both recurrence plots if the distance difference (see equation 4), has a value below a similarity threshold \( S_r \).

Figure 1. Stage one, normalization of the sequences of data to be compared.

Figure 2. Stage two, recurrence plots construction and comparison between them.
threshold and 0 above the threshold and are stored in a binary distance vector, a counting of the number of minimum differences (matches) is done, in order to determine the percentage of minimum distances between the compared recurrence plots, a set of five similarity percentages are established by the user, and depending on the corresponding percentage reached by the counting process a similarity category between the recurrence plots is determined. The Figure 2 shows the steps of this second stage.

Finally, in the third stage the visualization of the recurrence plots and their results are shown, the recurrence plots are visualized using a gray scale that corresponds with the distance between their points, the white color represents the minimum distance and the black color corresponds to the maximum distance present in the plot, each graphic of a recurrence plot is generated with a sampling of their corresponding distance set in order to facilitate its visualization, this is due to the enormous quantity of data generated, for example for a sequence of 1000 data a matrix of $10^6$ points is generated. In this stage, the option to save the recurrence plots and the results of their comparison is activated, multiple comparison experiments can be done in one run, for example: comparing a specific recurrence plot against a set of different recurrence plots can be saved by incremental storage of each new experiment, in the Figure 3 is showed the steps of this last stage.

![Figure 3. Stage 3, visualization of results.](image)

4. Experimental results

The technique was evaluated with a set of synthetic data sequences previously classified by their similarity degree by means of the direct comparison of the sequences using the technique of Derivative Dynamic Time Warping (DDTW) [5, 15]. Examples of different comparisons are showed in the screenshots of Figure 4 and Figure 5; in these examples a recurrence plot from a sequence tagged DS11 is compared with the corresponding recurrence plots for the sequences DS12 and DS42; in the previous classification the similarity of these sequences groups DS11 and DS12 in the same class and the sequence DS42 belongs to another class. The results with RecurrenceVs show a correspondence with the aforementioned results.

The experimental set contains a total of 55 synthetic data sequences that belong to 6 different classes that were previously assigned in [5, 15], in order to validate the proposed technique, randomly from each class a sequence was selected and compared with all the set, in this way a new classification tree was generated and compared with the classification tree reported in [5, 15], so we identified what differences result from the analysis of similarity between the sequences in a recurrence plot representation with respect to their direct or raw data representation. In Figure 6 is show, the behavior of the similarity percentage for the similarity analysis made with the RecurrenceVs tool, for the sequence DS50 that belongs to the class 1 in the DDTW classification. In Figure 7 is show, the behavior of the similarity percentage for the sequence DS12 that belongs to the class 5 in the DDTW classification.

In the six cases developed, the similarity analysis with recurrence plots groups the sequences in their corresponding classes (two examples are the Figure 6 and Figure 7) as in the DDTW classification, so the change in representation preserve the information about the general similarities in structure between sequences from the same family or class, but because we are comparing in a new representation as recurrence plots, the similarity at the level of complete families is different, this was observed when the classification tree was built with data of a matrix of average similarity percentages for each family with respect to the representative sequence used as reference (DS8, DS12, DS29, DS37, DS50, DS53), the Figure 8 shows the classification tree derived from the analysis with RecurrenceVs and the Figure 9 shows the corresponding tree derived from the DDTW classification in references [5, 15]. These differences in the similarity between families correspond with the new information about the data sequences structure extracted with the recurrence plot patterns.
Figura 4. Screenshot of the similarity analysis for the recurrence plots of two synthetic data sequences DS11 and DS12, their similarity corresponds to category 1 (95% of similarity).

Result: Category 1; 96.2334%

Figure 5. Screenshot of the similarity analysis for the recurrence plots of two synthetic data sequences DS11 and DS42, their similarity corresponds to category 3 (84.6% of similarity).

Result: Category 3; 84.6073%
Figure 6. Behavior of the similarity percentage between the recurrence plots for the data sequence DS50 and the rest of the sequences from the experimental set.

Figure 7. Behavior of the similarity percentage between the recurrence plots for the data sequence DS12 and the rest of the sequences from the experimental set.

Figure 8. Classification tree derived from the similarity analysis of recurrence plots of data sequences made with RecurrenceVs.

Figure 9. Classification tree derived from the similarity analysis of data sequences made with the DDTW in references [5, 15].
5. Conclusions

A computational technique for the analysis of similarity of data sequences using a new representation with recurrence plot patterns was presented, in this representation hidden patterns that represent the dynamics of a sequence can be visualized, analyzed, and compared; so new classifications based on these patterns that represent relations between dynamics and structure of the sequences can be built. The experiments of validation show expected changes in the way that the families or classes of sequences are grouped by their similarity due to the new representation with these patterns, but at the level of similarity between sequences of the same class the similarity is preserved when is compared with the classification method selected as our reference: DDTW. A current work is the application of RecurrenceVs for the comparison of genomic sequences; also we are working on the study of time series and their dynamics. Finally, new improvements of the functionality of the tool are on the way, and it will be posted in our website for free distribution under the GNU General Public License.

Acknowledgments. The first author thanks the financial support from PROMEP-SEP under the project of generation and application of knowledge UNACAR-PTC-085.

6. References


