An Empirical Comparison of Dissimilarity Measures for Time Series Classification

Conference Paper · October 2013

DOI: 10.1109/BRACIS.2013.22

2 authors:

Rafael Giusti
Federal University of Amazonas
17 PUBLICATIONS **112** CITATIONS
SEE PROFILE

Gustavo Enrique Batista
UNSW Sydney
122 PUBLICATIONS **5,029** CITATIONS
SEE PROFILE

2013 Brazilian Conference on Intelligent Systems

An Empirical Comparison of Dissimilarity Measures for Time Series Classification

Rafael Giusti, Gustavo E. A. P. A. Batista
Instituto de Ciências Matemáticas e de Computação
Universidade de São Paulo
São Carlos, SP, Brazil
rgiusti,gbatista@icmc.usp.br

***Abstract*—Distance and dissimilarity functions are of undoubted importance to Time Series Data Mining. There are literally hundreds of methods proposed in the literature that rely on a dissimilarity measure as the main manner to compare objects. One notable example is the 1-Nearest Neighbor classification algorithm. These methods frequently outperform more complex methods in tasks such as classification, clustering, prediction, and anomaly detection. All these methods leave open the distance or dissimilarity function, being Euclidean distance (ED) and Dynamic Time Warping (DTW) the two most used dissimilarity measures in the literature. This paper empirically compares 48 measures on 42 time series data sets. Our objective is to call the attention of the research community about other dissimilarity measures besides ED and DTW, some of them able to significantly outperform these measures in classification. Our results show that Complex Invariant Distance DTW (CIDDTW) significantly outperforms DTW and that CIDDTW, DTW, CID, Minkowski L-p (p-norm difference with data set-crafted "p" parameter), Lorentzian L-infinity, Manhattan L-1, Average L-1/L-infinity (arithmetic average), Dice distance, and Jaccard distance outperform ED, but only CIDDTW, DTW, and CID outperform ED with statistical significance.**

***Keywords*-dissimilarity measure; time series; classification;**

I. Introduction

In the last years, the Data Mining community has witnessed a huge increase of interest for time series methods and algorithms [1]. Such interest is justified by the innumerous applications that generate data across time. Virtually every piece of information collected from human, natural, and biological processes is susceptible to changes over time. And the study of how these changes occur is oftentimes a central issue to fully understand such processes.

Data Mining has contributed to the analysis of time series with a plenitude of methods for classification, clustering, motif discovery, anomaly detection, and time series prediction, among other tasks [2]. And for all the aforementioned tasks, a dissimilarity (or similarity) function frequently plays a central role in the method computation. For instance, dissimilarity measures are used in classification to assign the class mode among the most similar instances to a query example, a procedure known as k-nearest neighbor rule (k-NN). Though simple, the k-nearest neighbor strategy provides very competitive results, frequently outperforming more complex methods, such as [3]. In clustering, most procedures including partitional, hierarchical, and spectral clustering rely on a dissimilarity function to measure the similarity among objects. In the case of time series, recent work suggests that the choice of clustering algorithm is much less important than the choice of dissimilarity measure used, with Dynamic Time Warping providing excellent results [4]. In anomaly detection, dissimilarity functions are frequently used to detect anomalous instances. An instance too distant from every other instance in the data set is considered anomalous [5]. A recent survey on anomaly detection in time series has shown that similarity-based methods provide the best overall results [6].

In this work we conduct an experimental comparison of dissimilarity and similarity measures for time series. In order to have an objective measure to assess and compare our results, we have restricted our analysis to classification problems, due to the presence of a ground-truth. We have evaluated 48 dissimilarity and similarity functions surveyed in [7] augmented with DTW [8], a popular distance for time series, and CID and CIDDTW [9], two complexity-invariant distances. We have evaluated the measures in 42 publicly available benchmark data sets from the UCR Time Series Classification/Clustering Page [10].

Our results show that CIDDTW significantly outperforms DTW and that CIDDTW, DTW, CID, Minkowski L_p, Lorentzian L_∞, Manhattan L_1, Average L_1/L_∞, Dice, and Jaccard outperform ED, but only CIDDTW, DTW, and CID outperform ED with statistical significance.

The remainder of this paper is organized as follows: Section II briefly overviews our results as an introduction to the next sections. Section III presents and discusses the dissimilarity functions that outperform the Euclidean distance. Section IV details our experimental methods. Section V discusses our results in depth. And Section VI draws some conclusions and presents potential future work.

II. Results Overview

In this section we present an overview of our results. This is certainly a unusual paper organization. However, in lieu of including several pages long tables of experimental data, we have decided not to describe in details the results for all 48 measures we have evaluated. Thus, we show average (over all data sets) results for all measures in this

978-0-7695-5092-3/13 $26.00 © 2013 IEEE
DOI 10.1109/BRACIS.2013.22

section and select a subset of 9 measures that performed better than the Euclidean distance for further discussion. The Euclidean distance is a widely used measure for time series classification due to its simplicity, computational efficiency, and good empirical results [3]. We should note here that all discarded dissimilarity measures have $O(m)$ computational complexity, where m is the length of the time series. Therefore, the discarded functions provided worse classification performance than the Euclidean distance and required similar processing power.

Before we continue, the reader may have noted that we often use the term “dissimilarity measures” instead of “distance functions”. From the 38 dissimilarity measures we analyse in this work, only 5 are guaranteed distance functions, in the sense that they respect the requirements for determining a metric space. Therefore, we refer to those measures as “dissimilarity measures” for the sake of simplicity. Since the linear scan version of k-nearest neighbor does not require the properties of metrics, we can safely use these measures for classification.

We have also analyzed 10 similarity measures. A similarity measure gives a score that describes how similar two objects are, in contrast with distances and dissimilarity measures, which give a score describing how much two items differ. One should be only concerned that, when classifying with the nearest neighbor approach, dissimilarities should be minimized whilst similarities should be maximized.

Our study started with 45 similarity and dissimilarity measures surveyed in [7]. The original survey was intended for comparing probability density functions represented as histograms. Such histogram $H(X)$ has certain properties that time series do not cope with. For instance, the summation of all values $H_i(X) \in H(X)$ equals 1, and every value in the histogram fits the interval $[0, 1]$. Nevertheless, we successfully adapted each measure to our experiments with time series and employed all of them in 1-nearest neighbor classification. When two measures were equal or proportional one to another, we excluded one of them. We then augmented the initial set of measures to include DTW, CID, and CIDDTW.

In Table I we present the mean accuracy rates for all 42 data sets, as well as the standard deviations for the estimation of that statistics for each distance measure analyzed in this paper. Out of the 48 measures analysed, only 9 of them outperformed the Euclidean distance considering the average accuracies. These measures will be further described and analyzed in the next sections.

III. Selected Dissimilarity Measures

Because the Euclidean distance function is so prominently used for classification with the k-NN classifier, we have decided to use it as a baseline. Therefore, we have decided to not include in this section the description of any dissimilarity measure that under-performed the Euclidean distance.

Table I
Mean accuracy rates over all data sets

Measure	**Mean**	**Std. Deviation**
CIDDTW “Best Window” dissim.	0.8184	0.1374
DTW ”Best Window“ dissim.	0.8103	0.1419
DTW “Full Window” dissim.	0.7905	0.1519
CID Euclidean dissim.	0.7774	0.1460
Minkowski L$_p$ distance	0.7660	0.1417
Lorentzian distance	0.7640	0.1484
Manhattan (City Block) distance	0.7627	0.1501
Avg L_1/L_∞ distance	0.7619	0.1500
Dice dissim.	0.7529	0.1479
Jaccard dissim.	0.7529	0.1479
Euclidean distance	**0.7507**	**0.1493**
Chebyshev L_∞ distance	0.6719	0.1892
Hellinger dissim.	0.6408	0.1932
Kumar Johnson dissim.	0.5437	0.2556
Divergence dissim.	0.5128	0.1989
Soergel dissim.	0.4963	0.1843
Emanon 2 dissim.	0.4902	0.2232
Bhattacharyya similarity	0.4538	0.2912
Inner Product similarity	0.3586	0.3091
Dice similarity	0.3496	0.3135
Kumar similarity	0.3496	0.3135
Cosine similarity	0.3468	0.3133
Intersection similarity	0.3414	0.3021
Fidelity similarity	0.3310	0.2816
Clark dissim.	0.2938	0.2296
Kulczynski dissim.	0.2905	0.2304
Motyka dissim.	0.2778	0.2461
Sørensen dissim.	0.2778	0.2461
Max Symmetric χ dissim.	0.2720	0.2145
Pearson dissim.	0.2698	0.2033
Jensen Difference dissim.	0.2665	0.2503
Topsøe dissim.	0.2664	0.2503
Jeffrey dissim.	0.2659	0.2480
Kulczynski similarity	0.2653	0.2098
Square Chord dissim.	0.2647	0.2516
Wavehedges dissim.	0.2580	0.2119
Tanimoto dissim.	0.2561	0.2440
K Divergence dissim.	0.2503	0.2084
Motyka similarity	0.2503	0.2223
Vicis Wave Hedges dissim.	0.2501	0.2262
Squared χ^2 dissim.	0.2417	0.2255
Additive Symm χ^2 dissim.	0.2409	0.2184
Canberra dissim.	0.2408	0.2093
Neyman dissim.	0.2394	0.2264
Kullback dissim.	0.2394	0.2080
Taneja dissim.	0.2390	0.2150
Emanon 3 dissim.	0.2377	0.2298
Min Symmetric χ dissim.	0.2371	0.2213
Harmonic Mean similarity	0.2298	0.2188

The reader may refer to [7] for equations of all measures discussed in this paper.

DTW is a widely known algorithm for pattern matching which was introduced to the temporal data community in 1994 [8]. Though its implementation may involve some caveats, intuitively DTW is quite simple. It attempts to match two time series by “stretching” and “contracting” subsequences of the series so the “height” difference between the series is minimized. This idea is depicted in Figure 1. The figure shows both the original time series and an exaggerated separation of them to highlight the matching. Each line connecting the series is a matching between two

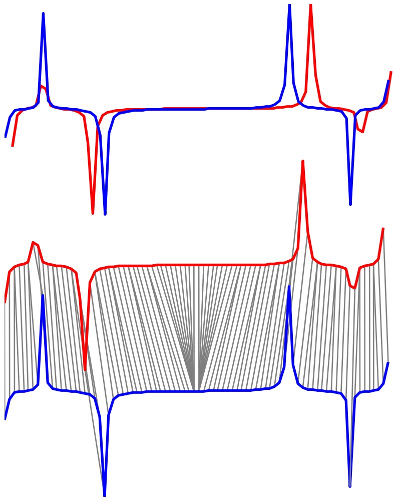

Figure 1. Dynamic Time Warp: two time series of approximately equal mean value (top) are exaggeratedly separated and the warping calculated by DTW is displayed (bottom)

observations. The DTW dissimilarity between the series is the square root of the sum of the differences between the actual matched observations.

The original DTW algorithm was parameter-free and was allowed to "stretch" the patterns as much as required. A more constrained version of the DTW incorporates the window size parameter δ. The δ parameter limits the number of observations a matching can occur ahead or behind any given observation. Oftentimes, this leads to improvement of the classification accuracy, since the constraint avoids pathological warpings [11]. We have run DTW both without the δ parameter and with certain selected values (how we chose these values will be discussed in the next section). We have identified these executions as "DTW Full window" and "DTW Best window".

A previous work [9] has noted that complex time series are frequently considered more similar to simple time series than to their actual nearest neighbors. The Euclidean distance and even the DTW dissimilarity may report false neighbors when used for k-NN classification if the complexity of the classes varies too much within a data set. For instance, a higher complexity class may have its objects frequently misclassified as a simpler class. In [9], a Complexity Invariant Distance (CID) approach is proposed. CID is a factor of difference of complexity between two time series. It is calculated as the estimated complexity of the more complex series divided by the estimated complexity of the less complex series. When two time series are similar in complexity (*i.e.*, both are very complex or both are not very complex), the CID factor tends to 1. When the two time series are not similar in complexity however (*i.e.*, one is very complex and one is not very complex), the CID factors tend to a number higher than 1. By multiplying the CID factor to the calculated dissimilarity, one may "punish" matchings between series that differ largely in complexity.

The complexity of a time series may be estimated by different means [12]. In our experiments, we have used the simple approach proposed in [9]. This approach is based on the concept that if a time series is "stretched" to the point where it is "flat", "complex" time series of equal length as "simple" time series will turn out longer, as illustrated in Figure 2.

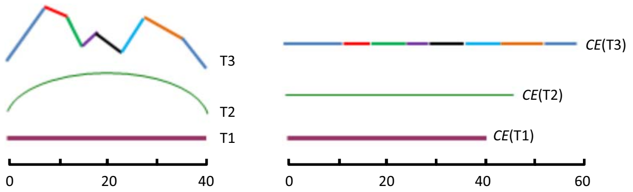

Figure 2. Estimation of the complexity of three time series. The least complex of them (T1) "stretches" to the shortest line segment. The most complex of them (T3) "stretches" to the longest line segment. Originally featured in [9]

For the remainder of this paper, let a time series of length m be an ordered sequence $Z = (Z_1, \ldots, Z_m)$, $Z_i \in \mathbb{R}$ for all $i \in [1, m]$. Given a time series Z, its complexity as previously explained may be numerically expressed as in Equation 1.

$$CE(Z) = \sqrt{\sum_{i=1}^{m-1} (Z_i - Z_{i+1})^2} \tag{1}$$

The Minkowski distance function is a generalization of the Euclidean distance function. It may be used to describe a family of distance functions differing one from another only by the value of the parameter p, which is described in Equation 2. When $p = 2$ it describes the Euclidean distance, which expresses our commonsense notion of "distance" measured as the length of a straight line between two points. When $p = 1$ we have the Manhattan distance function. This distance function is also called "city block distance" because it brings the notion of distance one would intuitively think of when walking in a city. Other values of p are much less intuitive but nevertheless perfectly plausible. The Chebyshev distance function may be considered an extreme case where $p \to \infty$ and is expressed by Equation 3.

$$d_{Mink}(S, Z) = \sqrt[p]{\sum_{i=1}^{m} |S_i - Z_i|^p} \tag{2}$$

$$d_{Cheb}(S, Z) = \max_{i=1}^{m} |S_i - Z_i| \tag{3}$$

One may also think of averaging distance functions. For instance, the average of the Manhattan distance with the Chebyshev distance yields surprisingly good results. We have identified this distance function as "Average L_1/L_∞".

Among dissimilarity functions unrelated to the Minkowski family, we have selected the Lorentzian (Equation 4), Jaccard (Equation 5), and Dice (Equation 6) dissimilarities.

$$d_{Lor}(S,Z) = \sum_{i=1}^{m} ln(1 + |S_i - Z_i|) \tag{4}$$

$$d_{Jac}(S,Z) = \frac{\displaystyle\sum_{i=1}^{m}(S_i - Z_i)^2}{\displaystyle\sum_{i=1}^{m}(S_i^2 + Z_i^2 - S_i Z_i)} \tag{5}$$

$$d_{Dic}(S,Z) = \frac{\displaystyle\sum_{i=1}^{m}(S_i - Z_i)^2}{\displaystyle\sum_{i=1}^{m} S_i^2 + \sum_{i=1}^{m} Z_i^2} \tag{6}$$

Several other dissimilarity measures were employed in our experiments. Though these measures are not individually discussed in this section, they are nonetheless listed in Table I. For reference on their formulas and discussion on some of their properties, please refer to [7].

IV. Experiments

In order to verify how the dissimilarity measures contrast to each other, we have conducted an extensive set of experiments on 42 data sets from the UCR Time Series Classification/Clustering Page repository [10]. The UCR repository is a comprehensive repository of time series including real and synthetic data sets. Currently, it is arguably the largest public repository of labeled time series data sets [1], encompassing data sets from a variety of domains.

The UCR repository data sets are shipped with a preset two-fold partition of training and testing data sets. Most data sets may be considered small or medium-sized, but none of them is partitioned with less testing instances than training instances. It is common that the proposed partition is respected by paper authors to encourage experiment reproducibility and comparison of results.

The experimental method we have followed to assess each measure consists of multiple executions of the 1-NN classification algorithm over the several data sets. The 1-NN classifier is a simple instance-based classifier that depends heavily on the similarity or dissimilarity measure employed and is also understood to be extremely competitive with more robust, complex classification models. Specifically, Xi et al. [13] claim that, when associated with DTW, the 1-NN classifier is "exceptionally difficult to beat". Such reasons make a good case for using the 1-NN to evaluate the efficacy of dissimilarity/similarity measures in classification tasks.

For sake of clarity, we have included in Figure 3 the 1-NN algorithm used in this paper to classify and estimate accuracy. We have repeated this process for every data set and measured the estimated accuracies, and used this statistic as an assessment of the distance measure.

```
Input: Training data set $S$, testing data set $T$, (dis)similarity measure $M$
Output: Estimated accuracy $Acc$
$Matches \leftarrow 0$
for each $Z_t \in T$ do
    find $Z_s \in S$ that minimizes/maximizes $M(Z_t, Zs)$
    if the class labels of $Z_s$ and $Z_t$ are the same then
        $Matches \leftarrow Matches + 1$
    end if
end for
$Acc \leftarrow \frac{Matches}{|T|}$
return $Acc$
```

Figure 3. Accuracy estimation for a given measure M

Two of the dissimilarity measures from our study rely on parameters: DTW "Best window" algorithm has the δ parameter and the Minkowski distance function has the p parameter. There are possibly optimal values of δ and p for each data set, but it is not obvious what these values are. We have used the algorithm presented in Figure 3 as an underlying base for the algorithm presented in Figure 4 to tune the p parameter for each data. Since we could not possibly test every single possible value of p, we first defined a set of sensible values for consideration. Then, when testing against a data set, we ran multiple executions of the 1-NN

```
Input: Training data set $S$, (dis)similarity measure $M$, set of parameter values $P$
Output: The selected value for the parameter
$BestValue \leftarrow \varnothing$
$BestAcc \leftarrow 0$
for each $p \in P$ do
    $SumOfAccs \leftarrow 0$
    for each instance $s \in S$ do
        $S_t \leftarrow \{s\}$
        $S_s \leftarrow S \setminus S_t$
        $Acc \leftarrow$ the estimated accuracy of 1-NN with (dis)similarity measure $M$ over the training data set $S_s$ and the testing data set $S_t$
        $SumOfAccs \leftarrow SumOfAccs + Acc$
    end for
    $MeanAcc \leftarrow \frac{SumOfAccs}{|S|}$
    if $MeanAcc > BestAcc$ then
        $MeanAcc \leftarrow BestAcc$
        $BestValue \leftarrow p$
    end if
end for
return $BestValues$
```

Figure 4. Parameter values estimation for distance measure d

classification algorithm against the training data set with a *leave-one-out* approach, experimenting different parameter values from that set in each run. The parameter value that provided the best accuracy in the training data set was used to assess the measure against the testing data set.

We have scanned the Minkowski distance function with values of p in the intervals $0.1, 0.2, 0.3, \ldots, 1.9$ and $3, 4, 5, \ldots, 15$. As for DTW, we have used for each data set the values of δ reported in [10]. We are comfortable in using these values because they are guaranteed to have been estimated on the training sets only.

V. Results Discussion

Since we have chosen the Euclidean distance as a baseline for our evaluation, it makes sense to compare all measures against it. Table II presents results for the top 14 measures and the Euclidean distance. Because merely comparing the mean accuracies between two algorithms can be misleading, we have conducted a Wilcoxon signed-rank paired test [14] between each of these measures and the Euclidean distance. Surprisingly, only the Dice and the Jaccard dissimilarity measures were not accused by our experiments to differ with statistical significance from the Euclidean distance – in which case we can not really claim their performance to be better or worse. For each of the other measures, we can claim, with $p = 0.05$ confidence, that they produced either better or worse results than the Euclidean distance.

We can also show that there are significant differences between other pairs of measures. The scatter plot in Figure 5 compares the accuracies obtained with DTW when using the δ parameter ("Best window") and when not using the δ parameter ("Full window"). Each point in the scatter plot represents a data set. A point that falls in the shadowed area indicates that the accuracy of DTW is higher when the δ parameter is used as explained in Section IV. The figure shows clearly a predominance of points in the shadowed area, indicating that a good choice of the δ parameter improves the classification accuracy. As a side advantage, because the δ parameter restricts the algorithm's search space, it also makes the classification faster.

Table II
Comparison of the top 14 measures and Euclidean distance

Measure	Mean	Stat. Significant?
CIDDTW "Best Window" dissim.	0.8184	Better
DTW "Best Window" dissim.	0.8103	Better
DTW "Full window" dissim.	0.7905	Better
CID Euclidean dissim.	0.7774	Better
Minkowski L_p distance	0.7660	Better
Lorentzian distance	0.7640	Better
Manhattan (City Block) distance	0.7627	Better
Avg L_1/L_∞ distance	0.7619	Better
Dice dissim.	0.7529	No
Jaccard dissim.	0.7529	No
Euclidean distance	**0.7507**	–
Chebyshev L_∞ distance	0.6719	Worse
Hellinger dissim.	0.6408	Worse
Kumar Johnson dissim.	0.5437	Worse
Divergence dissim.	0.5128	Worse

The same scatter plot approach can be used to compare DTW with CIDDTW. According to Figure 6, our experiments have revealed that CID increases the classification accuracy of the DTW dissimilarity measure with our δ selection approach. We have also run a Wilcoxon signed-rank paired test between the two measures, which accused statistically significant difference between them with $p = 0.05$ confidence.

But how do all measures compare to each other when considered altogether? One must refrain from the impulse of running multiple executions of a two-paired statistical test between all pairs of measures. More adequate statistical tests, such as the Friedman test, are available for comparing several algorithms over multiple data sets. However, the

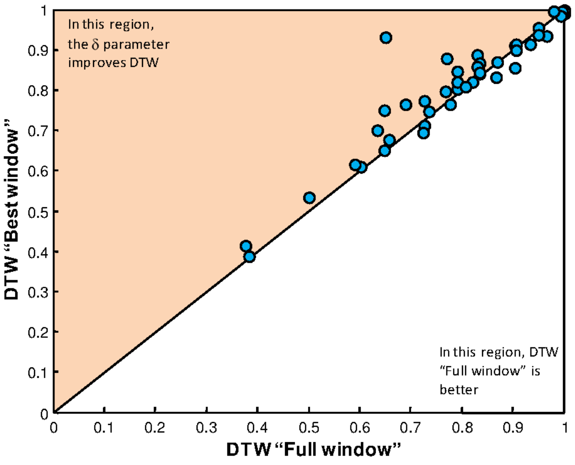

Figure 5. Scatter plot for the accuracy of restricted DTW against non-restricted DTW over all data sets

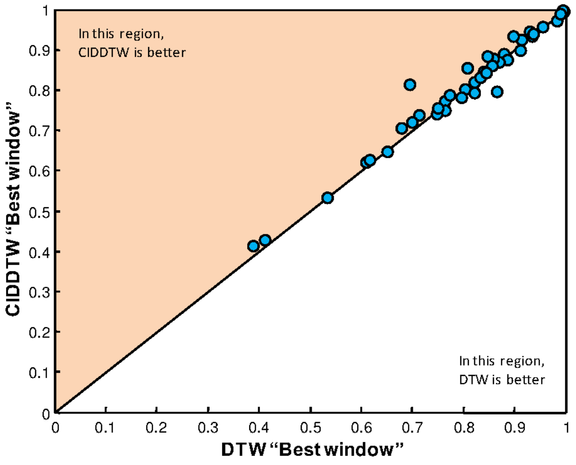

Figure 6. Scatter plot for the accuracy of the CIDDTW against DTW over all data sets

"power" of the statistical test depends both on the number of algorithms and data sets. The larger the number of algorithms, the larger the number of required data sets.

We included in the test the measures that were found to be significantly better than the Euclidean distance, except for DTW "Full window". We have executed a Friedman hypothesis test with $p = 0.05$ considering 8 measures over 42 data sets. After the Friedman test rejected the null hypothesis that all measures were equally comparable, we proceeded with a post-hoc Nemenyi test. The result of the Nemenyi test is presented as a critical distance graphic in Figure 7. The scale in the figure indicates the average rank of each measure. Sets of measures connected by a thick line have not presented statistically significant difference.

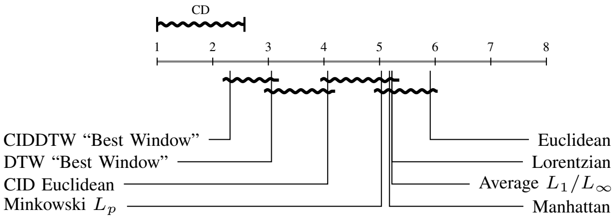

Figure 7. Nemenyi test result. Measures connected by a thick line have not presented statistically significant difference

It is interesting to note that the Nemenyi test did not accuse significant difference between CID and DTW (both with best window). The reader might find such a result contradicts the result of the Wilcoxon test. Actually not. Since the Nemenyi performs multiple comparisons, it requires more evidence in order to detect significant difference. It is worth recalling that the result of the Nemenyi test is not stating that there is no difference between CID and CIDDTW. Rather it is simply saying that there is not enough evidence to affirm that there is a statistically significant difference under the multiple comparison setting.

The Nemenyi test indicates that CIDDTW and DTW significantly outperform all other measures but CID. It seems that our experiments have a "take home" message that DTW-based measures should be preferred over the other measures for most applications. However, there is a caveat here. DTW-based measures are the only $O(m^2)$ measures in our experiments, where m is the length of the time series. The reader might be asking him/herself how these complexities will translate in terms of running time, and this question is not so easy to answer. The literature has several improvements proposed to speed up DTW calculations [15], [16], [1], and our implementation does not incorporate all these proposals. Nevertheless, despite our best effort to improve the speed of DTW-based measures, experiments with these measures took several times longer than all experiments with other measures combined.

VI. Conclusion

This paper empirically compares 48 dissimilarity measures in 42 time series data sets. Our objective is to call the attention of the research community about other dissimilarity measures besides ED and DTW, some of them able to significantly outperform these measures in classification. Our results show that CIDDTW significantly outperforms DTW and that CIDDTW, DTW, CID, Minkowski L_p, Lorentzian L_∞, Manhattan L_1, Average L_1/L_∞, Dice, and Jaccard outperform ED, but only CIDDTW, DTW, and CID outperform ED with statistical significance.

Given that DTW-based measures are the only $O(m^2)$ time complexity measures in our experiments, it is not surprising that its execution time has been superior to that of all other measures by several times. We defend that measures such as the relatively new CID Euclidean [9] and even the Minkowski L_p (which requires parameter tuning) deserve more attention from the data mining community.

As future work we intend to investigate the error correlation among these measures, so we can suggest possible measure compositions that will further improve the classification accuracy of the k-NN classifier.

Acknowledgment

We thank the Brazilian funding agency FAPESP for the financial support that made this research work possible – grant #2012/08923-8, São Paulo Research Foundation (FAPESP); and grant #2012/07295-3, São Paulo Research Foundation (FAPESP).

References

[1] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. Keogh, "Experimental comparison of representation methods and distance measures for time series data," *Data Mining and Knowledge Discovery*, vol. 26, pp. 275–309, 2013.

[2] C. M. Antunes and A. L. Oliveira, "Temporal data mining: an overview," in *KDD Workshop on Temporal Data Mining*, 2001, pp. 1–15.

[3] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, "Querying and mining of time series data: experimental comparison of representations and distance measures," *VLDB Endowment*, vol. 1, no. 2, pp. 1542–1552, 2008.

[4] Q. Zhu, T. Rakthanmanon, G. Batista, and E. Keogh, "A novel approximation to dynamic time warping allows anytime clustering of massive time series datasets," in *SDM*, 2012, pp. 999–1010.

[5] D. Yankov, E. Keogh, and U. Rebbapragada, "Disk aware discord discovery: finding unusual time series in terabyte sized datasets," in *ICDM*, 2007, pp. 381–390.

[6] V. Chandola, D. Cheboli, and V. Kumar, "Detecting anomalies in a time series database," Computer Science Department, University of Minnesota, Tech. Rep., 2009.

[7] S. Cha, “Comprehensive survey on distance/similarity measures between probability density functions,” *International Journal of Mathematical Models and Methods in Applied Sciences*, vol. 1, no. 4, pp. 300–307, 2007.

[8] D. Berndt and J. Clifford, “Using dynamic time warping to find patterns in time series,” in *Workshop on Knowledge Discovery in Databases*, 1994, pp. 359–370.

[9] G. Batista, X. Wang, and E. Keogh, “A complexity-invariant distance measure for time series,” in *SDM*, 2011, pp. 699–710.

[10] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana, “The ucr time series classification/clustering homepage: www.cs.ucr.edu/~eamonn/time_series_data/,” 2006.

[11] E. J. Keogh and C. A. Ratanamahatana, “Exact indexing of dynamic time warping,” *Knowledge and Information Systems*, vol. 7, no. 3, pp. 358–386, 2005.

[12] R. C. Prati and G. E. A. P. A. Batista, “A complexity invariant measure based on fractal dimension for time series classification,” *International Journal of Natural Computing Research*, 2013, in press.

[13] X. Xi, E. Keogh, C. Shelton, L. Wei, and C. A. Ratanamahatana, “Fast time series classification using numerosity reduction,” in *ICML*, 2006, pp. 1033–1040.

[14] J. Demšar, “Statistical comparisons of classifiers over multiple data sets,” *The Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[15] I. Assent, M. Wichterich, R. Krieger, H. Kremer, and T. Seidl, “Anticipatory dtw for efficient similarity search in time series databases,” *VLDB Endowment*, vol. 2, no. 1, pp. 826–837, 2009.

[16] T. Rakthanmanon, B. Campana, A. Mueen, G. Batista, B. Westover, Q. Zhu, J. Zakaria, and E. Keogh, “Searching and mining trillions of time series subsequences under dynamic time warping,” in *SIGKDD*, 2012, pp. 262–270.