A review and comparison of time series similarity measures

Maša Kljun[1], **Matija Teršek**[1], **Erik Štrumbelj**[1]

[1]*Faculty of Computer and Information science, University of Ljubljana*
E-mail: mk2700@student.uni-lj.si, mt2421@student.uni-lj.si, erik.strumbelj@fri.uni-lj.si

Abstract

We review 12 time series similarity measures and investigate their time complexity, normalization, invariance with respect to warping and scaling, support of time series of different lengths, and other properties. We show on simulated data that several similarity measures perform well on average, but none perform well in all cases and in some cases measures that typically perform poorly, such as compression-based similarity, are a better alternative.

1 Introduction

Measuring similarity between time series is an important component in time series data analysis, especially unsupervised learning. Many different measures exist and it is often not clear which measure is the best choice for the test at hand or how different measures compare with respect to relevant properties such as invariance to scaling/warping and time complexity.

The few related works are Wang et al. [13] who compare 9 measures but omit those based on correlation coefficients or compression. Serra and Arcos [10] and Górecki and Piasecki [6] compare 7 and 30 measures, respectively, but focus on 1-NN classification performance and not clustering performance and other properties as we do. Esling and Agon [4] do focus on other properties, but not on clustering performance.

We aim to provide a compact review and classification of the most commonly used similarity measures and relevant properties which are often excluded in related work. Furthermore, we use several simulated data sets to empirically evaluate how different measures compare to each other and how well they perform in clustering.

2 Distance measure features

In this paper we will view time series similarity measures in terms of these properties, which are relevant to choosing the best similarity measure for the task at hand:

- **Time complexity.**
- **Can compare time series of different lengths.**
- **Normalization.** Does increasing the length or sampling frequency of the time series, without changing any other properties, change the value? If so, we provide a factor that normalizes the measure and facilitates comparison across time series of different lengths.
- **Invariance/robustness with respect to warping and scaling.** Warping is a change of the time series' times that preserves the ordering. Scaling is multiplication of the time series' values with a constant. Related work is inconsistent about warping, so we additionally define *weak invariance* (the same change is applied to both compared time series) and *strong invariance* (the change is applied to only one of the time series). Strong invariance to warping implies weak invariance to warping.

A summary of similarity measures is in Table 1.

3 Distance measures

Let $X = x_1, ..., x_n$ and $Y = y_1, ..., y_m$ be the two time series whose similarity we are interested in. We also use X_{-n} and X_{-1} to represent X without the last and first point, respectively.

3.1 L_p norms/distances

Depending on the value of p, we have:

- **Manhattan** ($p = 1$): $\sum_{i=1}^{n} |x_i - y_i|$.
- **Minkowski** ($1 < p < \infty$): $\sqrt[p]{\sum_{i=1}^{n}(x_i - y_i)^p}$.
- **Euclidean** ($p = 2$): $\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$.
- **Infinite norm** ($p = \infty$): $\max_{i=1,...,n} |x_i - y_i|$.

In empirical evaluation, we use Euclidean distance.

3.2 DISSIM

DISSIM is designed to work with time series with different sampling rates. It is defined as the integral of the Euclidean distance between the two series, which are assumed to be linear between sampling points. It can be normalized by dividing by $(n - 1)$ when time series are extended, but no normalization is required if we increase the sampling frequency on the same time interval.

3.3 Dynamic Time Warping

Dynamic time warping (DTW) was introduced in order to overcome some of the restrictions of simpler similarity measures such as Euclidean distance. It is to this day one of the most popular measures.

The DTW costructs a mn matrix of squared distances between points of both time series, which is then used as a cost matrix when searching for the cheapest path between $(1,1)$ and (m,n). Path cost determines the similarity.

Normalization depends on the step pattern - allowed transitions and weights between matched pairs, when searching for an optimal path. Normalization is then made by dividing the distance by n, m or $n+m$ depending on the step pattern and slope weighting [5].

3.4 Edit distance on Real Sequences

The Edit distance algorithm is based on counting the number of insert, delete, and replace operations required to transform one string into another. It can be applied to time series where points from X and Y are considered a match if their absolute distance is less than some ϵ.

EDR is defined recursively as: $E(X,Y) =$

$$\min\{E(X_{-1},Y_{-1})+s, E(X_{-1},Y)+1, E(X,Y_{-1})+1)\},$$

where subcost s is 0 if elements match and 1 otherwise [3]. $E(X,Y)$ is 0 if either of the time series are empty. According to Chen et al. [3] we get the best clustering results, when ϵ is set to a quarter of the maximum standard deviation of time series.

3.5 Edit distance with Real Penalty

Edit distance with Real Penalty method (ERP) is also based on the Edit distance algorithm. When addition or deletion happens in Y, ERP treats this as a gap. The ERP distance is defined as $E(X,Y)$

$$= \begin{cases} \sum_{i=1}^{m} |x_i - g| & ; n = 0 \\ \sum_{i=1}^{n} |y_i - g| & ; m = 0 \\ \min\{E(X_{-1},Y_{-1}) + d(x_1,y_1), & \\ E(X_{-1},Y) + d(x_1,g), & \\ E(X,Y_{-1}) + d(y_1,g)\} & ; else \end{cases}, \quad (1)$$

with $d(x,y)$ being defined as a L_1 norm between the points x and y [2]. If one of them is a gap point, it is equal to a user defined constant g.

3.6 Longest common subsequence

We use the Longest common subsequence (LCSS) model to cope with various problems such as different sampling rates, different lengths, outliers, and efficiency [12]. It allows for unmatched elements and efficient approximate calculation. It is defined as

$$D(\delta,\epsilon,X,Y) = 1 - \frac{L_{\delta,\epsilon}(X,Y)}{\min(n,m)}, \quad (2)$$

where L is a function defined as $L_{\delta,\epsilon}(X,Y)$

$$= \begin{cases} 0, & A \\ 1 + L_{\delta,\epsilon}(X_{-n},Y_{-m}), & B \\ \max\{\, L_{\delta,\epsilon}(X_{-n},Y), L_{\delta,\epsilon}(X,Y_{-m})\}, & \text{else} \end{cases}, \quad (3)$$

where $A = \{n = 0 \vee m = 0\}$ and $B = \{|x_{xn} - y_{xm}| < \epsilon, |x_{yn} - y_{ym}| < \epsilon, |n-m| < \delta\}$.

3.7 TQuEST

This similarity measure starts by transforming each time series into disjoint intervals such that within every interval all the time series' values are above a threshold ϵ. Let $S(X,\epsilon)$ be the unique such transformation of X where the intervals are the largest.

The TQuEST measure is defined as:

$$T(X,Y) = \frac{1}{|S(X,\varepsilon))|} \sum_{s\in(X,\varepsilon))} \min_{s'\in S(Y,\varepsilon))} d(s,s') + \frac{1}{|S(Y,\varepsilon)|} \sum_{s'\in S(Y,\varepsilon)} \min_{s\in S(X,\varepsilon))} d(s',s) \quad (4)$$

where $d(a,b)$ is the Euclidean distance between two time intervals [1].

3.8 Cross-corelation

Cross-correlation (CCor) is based on the Pearson correlation coefficient [7]. It is defined as $d_{CC}(X,Y) = \sqrt{\frac{1-CC_0(X,Y)}{\sum_{k=0}^{ml} CC_k(X,Y)}}$, where CC_k is the lag-k covariance and ml is the maximum allowed lag between X and Y and should not exceed the length of the series. By default, it is $\min\{n,m\} - 1$ [7].

3.9 Compression-based similarity measure

The compression-based similarity measure (CDM) is a class of measures defined as $\text{CDM}(X,Y) = \frac{C(XY)}{C(X)+C(Y)}$, where $C(X)$ is the size in bytes of the compressed time series X and XY stands for concatenated time series X and Y. Any compression algorithm can be used for $C(\cdot)$. In the empirical evaluation, we use gzip.

3.10 Piccolo distance

This similarity measure was introduced by Piccolo [9] as a measure of similarity between two ARIMA processes. It is defined as the Euclidean distance of the coefficients of the series' $AR(\infty)$ formulations. The coefficients of the lower order series are padded with zeros to the length of the larger order. The Piccolo distance exist for any invertible ARIMA process [8].

3.11 Prediction-based distance

This is a class of similarity measures based on the idea that two time series are similar if they are close at a specific time point in the future. Vilar et al. [11] is an implementation of this idea where forecast densities at a specific point in the future $T + h$ are compared. The distance is then calculated as an indefinite integral of absolute difference between estimates of the forecast densities for time series X and Y at time $T + h$ [8]. We set the forecast horizon h to 1 in our empirical evaluation.

3.12 Embedding-based similarity

This class of measures is based on learning a vector representation of time series and then computing their similarity using a vector similarity measure, such as Euclidean distance. We implemented this idea using Euclidean distance and Random Warping Series (RWS) to create a vector representation of time series [14]. This method uses

the DTW between the given time series and the random time series distribution. Then a family of positive definite kernels can be constructed from a map given by DTW. Optimal are found using cross-validation.

The time complexity depends on the model used but is typically $O(n)$, excluding the time we require to learn the representation. It is normalized, weakly invariant to warping, but it is not invariant to scaling. In the empirical evaluation, we use an 8-dimensional representation (16 and 32-dimensions do not lead to better results).

4 Classification of similarity measures

Wang et al. [13] propose the following categories: lock-step, elastic, threshold-based, and pattern-based measures. Montero et al. [8] propose: complexity-based, prediction-based, model-free, and model-based measures. Esling and Agon [4] propose: shape-based, edit-based, feature-based, and structure-based measures. Each of these classifications provides a different but incomplete view of similarity measures. A reconciliation is beyond the scope of this paper but for the sake of completeness, we classify the measures used in this paper according to each of the three classifications (see Table 1).

5 Empirical evaluation and comparison

We generated 9 groups of 100 time series of length 100 (see Table 2). For each pair of groups we computed the similarity for each pair of time series. We then clustered them into 2 clusters with k-medoids clustering. We evaluated the clustering using the adjusted Rand index. We repeated this process for each similarity measure.

The purpose of this experiment was twofold. First, to identify which measures give similar values (see Figure 1). And second, to highlight scenarios where a similarity measure might fail and which alternative could be used (see Table 3 for a summary).

6 Discussion

While we did not cover all, we did cover the most popular similarity measures and at least one representative from each class of similarity measures, except Spatial Assembling Distance (SpADe). Several similarity measures perform well on average, but none perform well in all cases. In some cases less known measures like compression-based similarity are better, even though they typically perform poorly. Therefore, choosing the best similarity measure for the task at hand is not a trivial task and there is value in our review of their properties. Piccolo distance stands out as the only linear complexity similarity measure with good overall performance. Our embedding-based approach achieves similar performance and also has linear time complexity, excluding the time we require to learn the embedding. However, it was learned on labelled data and is not generally applicable. The results are consistent with the results from Wang et al. [13], Serra and Arcos [10], and Górecki and Piasecki [6], where DTW and Edit distance measures performed best although Serra and Arcos [10] and Górecki and Piasecki [6] report that advanced modifications of DTW outperformed other measures.

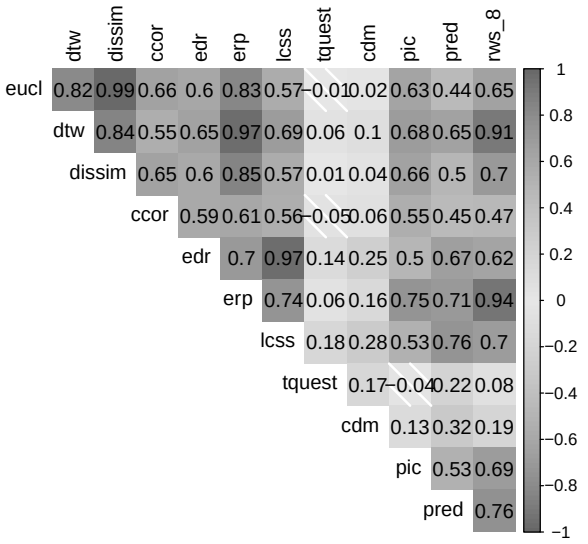

Figure 1: Pearson correlation between similarity measures across all pairs of time series from all nine groups. DTW, Euclidean distance, DISSIM, EDR, ERP, LCSS, and RWS all give numerically similar results. The following are particularly similar: Euclidean distance and DISSIM, DTW and ERP and RWS, and EDR and LCSS. CCor, PIC, and PRED are somewhat similar to each other and other methods. TQuEST and CDM strongly differ from the other measures and each other.

Further investigation of embeddings-based approaches is our main direction for further work. In particular, embeddings that are based on unlabelled data. While these approaches have been tremendously successful in image and video analysis and there have been notable applications in time series, there is no systematic treatment in the context of time series similarity and clustering. Finally, classification of time series similarities requires further work in order to reconcile the differences and inconsistencies between existing classifications and produce a more generally applicable classification.

References

[1] Aßfalg, J., Kriegel, H.-P., Kröger, P., Kunath, P., Pryakhin, A., and Renz, M. (2006). Tquest: threshold query execution for large sets of time series. In *International Conference on Extending Database Technology*, pages 1147–1150. Springer.

[2] Chen, L. and Ng, R. (2004). On the marriage of lp-norms and edit distance. In *Proceedings of the Thirtieth international conference on Very large data bases-Volume 30*, pages 792–803.

[3] Chen, L., Özsu, M. T., and Oria, V. (2005). Robust and fast similarity search for moving object trajectories. In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, pages 491–502.

	warping	scaling	Comp.	normalization	Wang	Esling	Montero	Diff. lengths
L_p	strong	no	$O(n)$	$\sqrt[p]{n}$	lock-step	shape	model-free	no
DISSIM	no	no	$O(n+m)$	$n-1$	lock-step	shape	*model-free*	no
DTW	strong	no	$O(nm)$	$n, m, n+m$	elastic	shape	model-free	yes
ERP	weak†	no	$O(nm)$	$max(n,m)$	elastic	edit	*model-free*	yes
EDR	weak†	no	$O(nm)$	$max(n,m)$	elastic	edit	*model-free*	yes
LCSS	weak†	no	$O(nm)$	normalized	elastic	edit	*model-free*	yes
Tquest	weak†	no	$O(nm \log nm)$	normalized	threshold	feature	*model-free*	yes
CCor	strong	yes	$O(nm)$	$(\sqrt{ml})^{-1}$	*elastic*	*feature*	model-free	yes
PIC	strong	yes	$O(n+m)$	$max(k1, k2)$	/	structure	model-based	yes
CDM	weak†	no	$O(n+m)^*$	normalized	/	structure	complexity	yes
PRED	no	no	$O(n+m)^*$	normalized	/	*structure*	prediction	yes*
RWS	weak	no	$O(n)^*$	normalized	/	*structure*	*model-based*	no

Table 1: Our classifications are in *italics*, / denotes that Wang et al. [13] treated different representations separately from similarity measures, $*$ denotes the most typical implementation but may vary with choice of underlying models, † denotes invariance based on the original definitions, where time is considered, but this vary with implementation.

group	type	noise
G1	-	N
G2	-	G
G3	linear	N
G4	linear	G
G5	linear + varying slope	N
G6	sine	N
G7	sine	G
G8	sine + varying phase	N
G9	sine + varying amplitude	N

Table 2: A summary of the 9 groups of simulated time series. N is normally distributed noise ($\mu = 0, \sigma = 0.5$), G is gamma-distributed noise ($\alpha = 0.5, \beta = 3$).

	mean	median	min
eucl	0.61	0.61	0.00
dissim	0.65	0.97	0.00
dtw	0.80	1.00	-0.00
edr	0.80	1.00	-0.00
erp	0.85	1.00	-0.00
lcss	0.83	1.00	-0.00
ccor	0.64	0.70	-0.00
cdm	0.21	0.04	-0.00
pic	0.85	1.00	0.01
pred	0.55	0.54	-0.00
tquest	0.36	0.19	-0.00
rws	0.87	1.00	0.25

Table 3: Aggregated adjusted Rand index. DTW, EDR, ERP, LCSS, CCor, PIC, and RWS are the best, on average, and CDM and TQuEST perform poorly. However, all similarity measures have poor worst-case performance - for each there is at least one pair of groups where it performs no better than random. CDM is the best at distinguishing between linear time series and linear time series with varying slope, where all other similarity measures perform poorly. Most measures also poorly distinguish between G1-G2 (normal noise vs gamma noise), G1-G5, and G2-G5 (normal or gamma noise vs linear time series with varying slope). Complete results can be found at `https://github.com/tersekmatija/ERK-2020`.

[4] Esling, P. and Agon, C. (2012). Time-series data mining. *ACM Computing Surveys (CSUR)*, 45(1):1–34.

[5] Giorgino, T. et al. (2009). Computing and visualizing dynamic time warping alignments in R: the dtw package. *Journal of statistical Software*, 31(7):1–24.

[6] Górecki, T. and Piasecki, P. (2019). A Comprehensive Comparison of Distance Measures for Time Series Classification. In *Workshop on Stochastic Models, Statistics and their Application*, pages 409–428. Springer.

[7] Liao, T. W. (2005). Clustering of time series data—a survey. *Pattern recognition*, 38(11):1857–1874.

[8] Montero, P., Vilar, J. A., et al. (2014). TSclust: An R package for time series clustering. *Journal of Statistical Software*, 62(1):1–43.

[9] Piccolo, D. (1990). A distance measure for classifying ARIMA models. *Journal of Time Series Analysis*, 11(2):153–164.

[10] Serra, J. and Arcos, J. L. (2014). An empirical evaluation of similarity measures for time series classification. *Knowledge-Based Systems*, 67:305–314.

[11] Vilar, J. A., Alonso, A. M., and Vilar, J. M. (2010). Non-linear time series clustering based on non-parametric forecast densities. *Computational Statistics & Data Analysis*, 54(11):2850–2865.

[12] Vlachos, M., Kollios, G., and Gunopulos, D. (2002). Discovering similar multidimensional trajectories. In *Proceedings 18th international conference on data engineering*, pages 673–684. IEEE.

[13] Wang, X., Mueen, A., Ding, H., Trajcevski, G., Scheuermann, P., and Keogh, E. (2013). Experimental comparison of representation methods and distance measures for time series data. *Data Mining and Knowledge Discovery*, 26(2):275–309.

[14] Wu, L., Yen, I. E.-H., Yi, J., Xu, F., Lei, Q., and Witbrock, M. (2018). Random warping series: A random features method for time-series embedding. *arXiv preprint arXiv:1809.05259*.